



TU Clausthal

Technische Universität Clausthal

Institut für Informatik - Software Systems Engineering

Lehrstuhl von Prof. Dr. Andreas Rausch

Masterarbeit

Entwicklung eines Mixed Reality Frameworks zur Objekterkennung in Industrie 4.0 Anwendungen

Sebastian Lawrenz

Matr.-Nr.: 419259

12.04.2018

Erstgutachter: Prof. Dr. Andreas Rausch

Zweitgutachter: Prof. Dr. Christian Siemers

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig angefertigt und alle Quellen und Hilfsmittel vollständig angegeben habe.

Hiermit erkläre ich mich einverstanden, dass meine Masterarbeit in der Instituts- und Universitätsbibliothek ausgelegt und zur Einsichtnahme aufbewahrt werden darf.

Clausthal-Zellerfeld, den 12.04.2018

(Sebastian Lawrenz)

Abstract

Der Übergang von Industrie 3.0 zu Industrie 4.0 hat die Rolle des Menschen im industriellen Produktionsprozess verändert. Im Rahmen von automatisierter Fertigung und Robotik begann der Ablöseprozess des Menschen bereits in der dritten industriellen Revolution. Mit der vierten rückt er gänzlich in den Hintergrund des Produktionsprozesses und übernimmt als Augmented Operator lediglich prozessüberwachende Aufgaben und Entscheidungen.

Diese veränderte Rolle des Menschen erfordert langfristig neue veränderte Mensch Maschinen Schnittstellen. Hierfür bieten sich insbesondere Mixed Reality Technologien an, welche eine Vermischung der physikalischen und digitalen Realität darstellen.

Eine Herausforderung beim Entwurf von Mixed Reality Anwendungen ist die Erkennung von beweglichen Objekten. Insbesondere wenn diese rein optisch nicht zu unterscheiden sind ist es eine große Herausforderung diesen die richtigen dazugehörigen Informationen zuzuordnen.

Um dieses Problem zu lösen wurde ein allgemeines Lösungspattern, genannt Position Mapper, entworfen. Dieses ermöglicht es mithilfe eines Tracking Verfahrens zunächst die Position eines Objektes im Raum zu ermitteln. Wenn sich das Objekt hierbei auf einer für den Position Mapper bekannten Strecke befindet, kann er mithilfe der aktuellen Informationen, Informationen über die Strecke, sowie dem bekannten Verhalten des Objektes, dieses eindeutig identifizieren.

Im Rahmen dieser Arbeit wurde dieses Pattern zur Evaluation prototypisch auf der Microsoft HoloLens implementiert. Diese trackt mithilfe eines visuellen Trackingverfahrens Werkstücke einer Modelfabrik. Die Werkstücke selbst können rein visuell nicht unterschieden werden, besitzen aber eine eindeutige Identität auf ihrem RFID Chip gespeichert ist. Die Modelfabrik verfügt über Sensorik um diese an bestimmten Punkten auszulesen und zwischen zu speichern. Der Position Mapper nutzt diese Informationen um ein erkanntes Werkstück eindeutig zu identifizieren.

Inhaltsverzeichnis

I.	Glossar	vi
1	Einleitung	1
1.1	Motivation	1
1.2	Ziele dieser Arbeit	3
1.3	Struktur dieser Arbeit	4
2	Grundlagen	5
2.1	Industrie 4.0	5
2.1.1	Definition und Zielstellung	5
2.1.2	Veränderte Aufgabenbereiche des Menschen im Kontext Industrie 4.0	8
2.2	Mensch Maschinen Schnittstellen Technologien	9
2.2.1	Klassische Industrie Devices und portable Devices	9
2.2.2	Head Mounted Display	10
2.3	Mixed-Reality	11
2.3.1	Virtual Reality	12
2.3.2	Augmented Reality	14
2.3.3	Zusammenfassung: Microsofts Mixed-Reality	19
2.3.4	Einsatzmöglichkeiten für Mixed Reality im industriellen Kontext	20
3	Problemstellung	24
3.1	Ausgangssituation des Beispiels	24
3.2	Übersicht	25
3.3	Physikalische Welt	26
3.4	Digitale Welt	27
3.5	Mixed Reality Welt	28
3.5.1	Simultaneous Localization and Mapping	28
3.6	Zusammenfassung	30

4	State of the Art	32
4.1	Augmented Reality in der Forschung	32
4.2	Objekterkennung	33
4.2.1	Detektion	34
4.2.2	Tracking	35
4.2.3	Nichtvisuelles-Tracking	38
4.2.4	Visuelles-Tracking	40
4.2.5	Identifikation	44
4.3	Informationsfusion.....	45
4.4	Zusammenfassung	47
5	Lösungskonzept.....	49
5.1	Position Mapper Pattern	50
5.1.1	Lösung.....	50
5.1.2	Struktur.....	50
5.1.3	Dynamik	51
5.1.4	Zusammenfassung.....	54
6	Implementierung	55
6.1	Szenario des Prototyps.....	55
6.2	Technologien	57
6.2.1	Unity.....	57
6.2.2	C#	60
6.2.3	Mixed-Reality Toolkit.....	60
6.2.4	Vuforia	62
6.2.5	Modelfabrik API	63
6.3	Implementierung des Prototyps	63
6.3.1	Modelfabrikinformationssystem	64
6.3.2	Sharing Service	65

6.3.3	Objekterkennung	66
7	Evaluation.....	68
7.1	Objekterkennung	68
7.1.1	Mithilfe der HoloLens	68
7.1.2	Visuell mithilfe von Vuforia	69
7.2	Lösungskonzept.....	72
8	Fazit und Ausblick	74
8.1	Fazit	74
8.2	Ausblick.....	74
9	Anhang	I
9.1	Abbildungsverzeichnis	I
9.2	Tabellenverzeichnis	II
9.3	Literaturverzeichnis	III

I. Glossar

Augmented Reality (AR): Bezeichnet eine computergestützte Wahrnehmung. Die physikalische Realität wird um virtuelle Inhalte erweitert, welche direkt auf die reale Umwelt abgebildet werden.

Detektion: Unter Detektion wird ein Verfahren verstanden, welches den Typen eines Objektes ermitteln kann, aber nicht seine Position.

Dynamische Objekte: Bezeichnen im Rahmen dieser Arbeit physikalische Objekte welche sich fortbewegen. Das heißt sie haben eine variable Position. Das Gegenteil sind statische Objekte. Diese haben eine unveränderliche Position.

Head Mounted Display (HMD): Bezeichnet ein am Kopf befestigtes Display. Insbesondere Datenbrillen wie die HoloLens, das EyeTab oder Google Glass werden als HMDs bezeichnet.

Head Up Display (HuD): Ein Anzeigesystem bei dem der Nutzer seine Blickrichtung beibehalten kann, da die Informationen direkt in sein Sichtfeld projiziert werden. Ein bekanntes Anwendungsfeld ist für Piloten. Im Gegensatz zu HMDs werden diese nicht direkt am Kopf befestigt und verfügen über keine weitere Sensorik.

Image Target: Bezeichnet in Vuforia ein visuelles Tracking Framework, welches Bilder bzw. Muster anhand ihrer spezifischen Merkmale erkennt.

Informationsfusion: Bezeichnet ein Verfahren in welchem durch die Kombination von unterschiedlichen Daten eine bessere präzisere Auswertung dieser erlangt wird.

Marker: Bei einem Marker handelt es sich um ein zwei- oder dreidimensionales Objekt, welches durch seine Form leicht visuell getrackt werden kann.

Mixed Reality (MR): Bezeichnet in der klassischen Definition das gesamte Realitäts-Virtualitäts-Kontinuum zwischen der physikalischen und der digitalen Realität. In neueren Definitionen und im Rahmen dieser Arbeit bezeichnet es insbesondere auch eine Augmented Reality welche mit der physikalischen Umwelt interagiert.

Mixed Reality Toolkit (MRT): Ein von Microsoft bereitgestelltes Open Source Framework, welches über eine Vielzahl von Lösungsmöglichkeiten, insbesondere für die HoloLens, und andere Mixed Reality Headsets verfügt.

Modelfabrik: Eine Lernfabrik welche zu Ausbildungs- und Forschungszwecken am Lehrstuhl für Software Systems Engineering der TU Clausthal zur Verfügung steht. Es handelt sich um die Nachbildung eines Hochregallagers und einer Produktionsstraße. Gesteuert wird die Modelfabrik von zwei Siemens S7-300.

Modelfabrik API: Eine Sammlung von in Java implementierten Methoden zur Kommunikation und zur Steuerung mit der Modelfabrik.

Object Target: Bezeichnet in Vuforia ein visuelles Tracking Framework, welches Objekte anhand der Merkmale eines vorgegebenen Objektes erkennt.

Objekterkennung: Im Rahmen dieser Arbeit wird unter Objekterkennung ein Verfahren verstanden, welches die Position und die eindeutige Identität eines Objektes ermittelt.

Pattern: Bezeichnet in der Softwaretechnik ein Analyse-, Entwurfs-, oder Architekturmuster. Sie stellen wiederverwendbare Vorlagen zur Lösung Problemlösung dar.

RFID: engl. für radio-frequency identification. Bezeichnet eine Technologie zur automatischen und berührungslosen Identifikation von Objekten

SLAM: engl. für Simultaneous Localization and Mapping bezeichnet ein ursprünglich aus der Robotik stammendes Problem, bei dem der Roboter zum einen eine Karte seiner Umgebung erstellen muss und zum anderen seine eigene Position innerhalb dieser Karte ermitteln muss. Im Rahmen dieser Arbeit ist dies für das MR-Device relevant.

SPS: Eine Speicherprogrammierbare Steuerung ist ein Gerät, welches zur Steuerung oder Regelung einer Maschine oder Anlage eingesetzt wird.

Tracking: Unter Tracking wird ein Verfahren verstanden, welches die Position eines Objektes im Raum ermittelt.

Unity: Der Name einer Laufzeit- und Entwicklungsumgebung für Spiele (auch als Spiele-Engine bezeichnet). Wird im Rahmen dieser Arbeit für die Entwicklung der MR-Anwendungen auf der HoloLens verwendet.

Visuelles Tracking: Ein Tracking Verfahren, welches auf optischen Erkennungsverfahren basiert.

Virtual Reality (VR): Bezeichnet die Darstellung einer computergenerierten 3D-Welt, welche eine für den Benutzer eine vergleichbare Wahrnehmung wie die physikalische Realität erzeugt.

Vuforia: Der Name eines AR-Frameworks, welches insbesondere visuelle Tracking Verfahren bereitstellt.

1 Einleitung

1.1 Motivation

Industrie 4.0 ist das dominierende Schlagwort der letzten Jahre. Seitdem der Begriff erstmalig auf der Hannover-Messe 2013 auftauchte, ist er aus dem Sprachgebrauch (zumindest in der Industrie und Forschung) nicht mehr wegzudenken. Zusammenfassend beschreibt Industrie 4.0 das Ziel, nach der dritten Industriellen Revolution, des Eintrittes der Digitalisierung und der Robotik im Industriebereich, die vierte industrielle Revolution einzuläuten. Ziele dieser sind unter anderem die vollständige Vernetzung, welche den Automatisierungsgrad soweit erhöht, dass der Mensch innerhalb des Produktionsprozesses keine tragende Komponente mehr sein wird, dies läuft unter dem Schlagwort Smart Factory¹. Mit Erreichen der autonomen Fertigung wird sich dementsprechend auch die Rolle, beziehungsweise der Arbeitsplatz, des Menschen grundlegend verändern. Die Rolle des einfachen Produktionshelfers rückt in den Hintergrund oder wird sogar vollständig wegfallen.

Diese Veränderungen werden zwangsweise dazu führen, dass sich die primären Aufgabenbereiche für den Menschen im Fertigungsprozess verschieben. Seine Rolle verschiebt sich vom aktiven Prozessteilnehmer zum passiven Beobachter und Entscheider (Augmented Operator) [RöB14]. Ein weiterer Aufgabenbereich ist die Wartung und Reparatur. Für beide Szenarien sind Mensch-Maschinen-Schnittstellen erforderlich. In der einfachsten Ausführung kann es sich hierbei lediglich um ein paar einfache Schalter zur Interaktion und einigen LEDs zur Zustandsvisualisierung handeln wohingegen in höheren Ausführungen Panel PCs oder anderen Devices eingesetzt werden, welche ausführlichere Informationen zu aktuellen Maschinenzuständen, Produktionsdaten, Handlungsanweisungen oder sonstige Daten enthalten.

Mit dem steigenden Grad der Autonomisierung erhöht sich auch das Informationsaufkommen und gerade bei autonomen sich bewegenden Komponenten (z.B. selbstfahrende Fahrzeuge in Fertigungshallen) kommt die Frage auf, wo die Informationen visualisiert werden sollen?

¹ Die Smart Factory bezeichnet in der Forschung die Vision einer vollständig autonomen Produktionsumgebung ohne jegliche menschliche Eingriffe [KaWH12].

Neben der ständigen Verbesserung der klassischen und portablen Devices, welche eine potenzielle Möglichkeit für die Informationsvisualisierung bieten, gab es in den letzten Jahren große Fortschritte im Bereich der Head Mounted Displays (kurz HMD), welche ebenfalls über ein großes Potential zur Informationsdarstellung verfügen. Hierbei handelt es sich um Datenbrillen, welche die virtuellen Informationen auf einem Bildschirm nahe dem Auge darstellen oder sogar direkt auf die Netzhaut projizieren. Einer Deloitte Studie zufolge gewinnen diese innerhalb der Industrie immer mehr an Bedeutung und deutsche Unternehmen werden bereits 2020 knapp 850 Millionen Euro für Anwendungen in diesem Bereich investieren [EsOL16].

Mit HMDs werden insbesondere die Technologien: Virtual Reality (VR) und Augmented Reality (AR) verbunden. Die VR beschreibt ein virtuelles 3D Abbild der Wirklichkeit oder einer imaginären Umgebung, welche dem Nutzer die gleiche Erfahrung wie die Realität bieten soll.

Bei der AR hingegen handelt es sich um eine Methode bei der virtuelle Objekte auf die reale Welt abgebildet werden. Hierbei wird die reale Umwelt mithilfe von Sensorik (z.B. durch eine Kamera) erfasst und das virtuelle Objekt auf die wahrgenommene Realität abgebildet. AR Anwendungen können beispielsweise auf Smartphones und Tablets verwendet werden, insofern diese über eine Kamera verfügen, oder auf den bereits erwähnten Datenbrillen, welche die AR-Objekte direkt auf einen transparenten Bildschirm auf Augenhöhe abbilden. Eine der ersten bekannteren Datenbrillen war die Google Glass im Jahr 2013, welche zwar nie offiziell erschienen ist, aber die HMDs mit AR-Technologien für die breite Masse bekannt machte. Anfang 2015 stieg auch Microsoft in den Markt ein und stellte die HoloLens, eine Datenbrille welche neben einer normalen Kamera noch über einen Tiefenscanner verfügt, um sich ein genaueres Abbild seiner Umwelt erstellen zu können. Diese wird zwar derzeit offiziell noch als Developer Gerät verkauft und ist nicht im freien Handel erhältlich, aber besitzt trotzdem bereits eine große Community und ist in Forschung und Industrie bereits sehr beliebt (siehe auch Kapitel 2.3.2.2).

Im Zusammenhang mit der HoloLens wird häufig der Begriff „Mixed Reality“ verwendet. Dieses Schlagwort, welches ebenfalls von Microsoft etabliert wurde, beschreibt eine AR, in welcher nicht einfach nur virtuelle Objekte in der realen Umwelt abgebildet werden, sondern auch mit dieser in Beziehung stehen und interagieren. Ein

einfaches Beispiel hierfür wäre ein AR Spiel, bei welchem ein virtueller Avatar innerhalb der realen Umwelt des Spielers gesteuert wird und beispielsweise durch diese begrenzt wird. Der Avatar² kann z. B nicht durch Wände oder Gegenstände hindurchgehen, sondern muss diese umrunden.

Die Mixed Reality stellt durch ihre Interaktion mit der realen Umwelt nicht nur ein erhebliches Potenzial im Spielbereich zur Verfügung, sondern auch für Enterprise Anwendungen im industriellen Umfeld.

Zusammenfassend lässt sich sagen, dass durch den steigenden Grad der Automatisierung der Mensch im Produktionsprozess immer weiter in den Hintergrund rückt und insbesondere die Rolle eines Beobachters übernimmt. Dazu kommen die Unmengen an Daten und Informationen die visualisiert werden sollen. Die Frage ist wie können diese Daten geeignet visualisiert werden?

Mögliche Lösungsansätze hierfür sind Datenbrillen und Mixed Reality Technologien, die daher im Rahmen dieser Arbeit betrachtet werden.

1.2 Ziele dieser Arbeit

Ziel dieser Arbeit ist der Entwurf eines Mixed Reality Frameworks, welches insbesondere eine Möglichkeit für Mensch Maschinen Schnittstellenlösungen bildet. Dabei wird insbesondere die Möglichkeit fokussiert die Informationen direkt an ihren dazugehörigen physikalischen Objekten zu visualisieren.

Hierfür sollen zunächst im Kontext Industrie 4.0 die Ursachen erörtert werden, warum neue Mensch Maschinen Schnittstellen erforderlich sind und anschließend die Grundlagen von Augmented- und Mixed Reality eingeführt werden. Anschließend werden sollen die Möglichkeiten der Objekterkennung diskutiert und eingeführt werden, sowie auf Basis dieser und auf Grundlage anderer bestehender Arbeiten ein theoretisches Konzept erstellt zur Objekterkennung von Werkstücken im industriellen Kontext entwickelt werden.

Primär soll also ein allgemeines Lösungspattern für die Möglichkeiten der Objekterkennung dynamischer (in diesem Fall beweglicher) physikalischer Objekte gefunden werden. Darüber hinaus soll dieses prototypisch anhand einer Modelfabrik am

² Bezeichnet eine künstliche Person oder eine Grafikfigur die von einem Benutzer gesteuert wird.

Lehrstuhl für Software Systems Engineering der TU Clausthal umgesetzt und evaluiert werden.

1.3 Struktur dieser Arbeit

Zu Beginn dieser Arbeit wird in Kapitel 2 Grundlagen zunächst die Einflüsse von Industrie 4.0 auf die Rolle des Menschen im industriellen Fertigungsprozess untersucht. Anschließend werden auf Basis dieser Ergebnisse Mensch Maschinen Schnittstellen erörtert. Insbesondere werden dabei die Möglichkeiten von HMDs und Mixed-Reality Anwendungen diskutiert.

In Kapitel 3 Problemstellung werden die Probleme herausgearbeitet, welche sich ergeben, wenn eine Mixed Reality Anwendung entwickelt werden soll. Insbesondere das Problem digitale Informationen an ihrem dazugehörigen (dynamischen) physikalischen Objekt korrekt zu positionieren.

Darauffolgend wird in Kapitel 4 State of the Art der aktuelle Stand der Forschung im Bereich Augmented Reality vorgestellt. Anschließend werden verschiedene Verfahren und Definitionen rund um das Thema Objekterkennung vorgestellt.

In Kapitel 5 Lösungskonzept wird anschließend auf Basis des aktuellen Stands der Forschung ein Konzept vorgestellt, welches Objekterkennung für dynamische Objekte in festgelegten Bereichen ermöglicht.

Auf Basis dieses Konzeptes wird im Anschluss als Proof of Concept die Implementierung eines Prototypens vorgestellt, welcher als MR Hardware die HoloLens von Microsoft verwendet und physikalische Objekte der Modelfabrik erkennt. Hierfür wird zu Anfang das Szenario vorgestellt, anschließend die zugrunde liegenden Technologien eingeführt und abschließend die Implementierung des Prototyps. All dies findet sich im Kapitel 6 Implementierung.

Anschließend wird in Kapitel 7 Evaluation eine Auswertung von der Stabilität unterschiedlicher Tracking Verfahren vorgenommen, welche im Prozess dieser Arbeit implementiert und getestet worden, sowie überprüft, ob sich das in Kapitel 5 vorgestellte Lösungskonzept zur Objekterkennung eignet.

Schlussendlich werden im letzten Kapitel 8 Fazit und Ausblick die Ergebnisse der Arbeit nochmal einmal Review passiert und ein Ausblick der Möglichkeiten für kommende Arbeiten und Verbesserungspotentiale gegeben.

2 Grundlagen

Ausgehend von der immer weiter zunehmenden Automatisierung bis hin zur autonomen Fertigung werden in diesem Kapitel zunächst die Grundlagen dieses Wandels und die Ideen des Forschungsfeldes Industrie 4.0 vorgestellt. Anschließend wird kurz auf die daraus resultierende veränderte Rolle des Menschen im Fertigungsprozess eingegangen, welcher zwar keine aktive Komponente innerhalb des Produktionsprozesses spielt, aber weiterhin im Hintergrund die Prozesse überwacht, Entscheidungen trifft, oder Wartungsarbeiten vornimmt.

Motivation dieser Arbeit ist es unter anderem unter Berücksichtigung aktueller Technologien, eine neue flexiblere Mensch-Maschinen Schnittstellen Lösungen für industrielle Anlagen zu entwickeln. Mixed-Reality Technologien bieten hierbei besonders interessante Lösungsmöglichkeiten, da diese nicht nur zur Informationsaufbereitung für den Menschen dienen können, sondern insbesondere auch mit ihrer physikalischen Umwelt interagieren. Deswegen werden diese zum Abschluss des Kapitels intensiver vorgestellt und die Einsatzmöglichkeiten von MR-Lösungen in industriellen Projekten erörtert.

2.1 Industrie 4.0

Der Markt hat sich in den letzten Jahren immer mehr vom Verkäufer- zum Käufermarkt hin gewandelt [BaHV14, S7]. Dadurch neigt sich die Massenproduktion in bestimmten Bereichen dem Ende zu. Der neue Trend sind personalisierte, auf den Käufer zugeschnittene Produkte. Dementsprechend müssen die Unternehmen sich diesem Wandel anpassen und ergo auch ihre Produktion Strategien anpassen. In Deutschland werden Strategien um auf diesen Wandel zu reagieren unter dem Begriff „Industrie 4.0“ erforscht und erarbeitet.

2.1.1 Definition und Zielstellung

Industrie 4.0 steht hierbei für die 4.0 industrielle Revolution. Der Begriff industrielle Revolution ist hierbei historisch gewachsen und wird insbesondere mit der Einführung der ersten mechanischen Produktionsanlagen (insbesondere des Webstuhles) in Großbritannien Ende des 18. Jahrhunderts verbunden.

Die zweite industrielle Revolution folgte ca. ein Jahrhundert später und wurde durch den Aufstieg der Chemie- und Elektrotechnikindustrie vorangetrieben. Insbesondere wurde

sie durch den Beginn der Massenproduktion geprägt. Insbesondere wird dies mit der ersten Fließbandfertigung bei Ford und dem Beginn des Taylorismus verbunden.

Die dritte industrielle Revolution wird auch als die digitale Revolution bezeichnet. Mit dem Aufkommen der Digitalisierung konnte der Grad der Automatisierung innerhalb der Industrie noch weiter erhöht werden. Prägend war neben der Einführung des Computers, insbesondere auch der zunehmende Einsatz von Robotik und Speicherprogrammierbaren Steuerungen (SPS).

Die Idee der vierten Industriellen Revolution setzt nahtlos an der Digitalisierung an. Die im Zuge der Digitalisierung installierten Komponenten, welche oft noch für sich alleine stehen, sollen miteinander vernetzt werden, um die Fertigung weiter zu individualisieren. Abbildung 1 fasst die vier Stufen der industriellen Revolution noch einmal zusammen

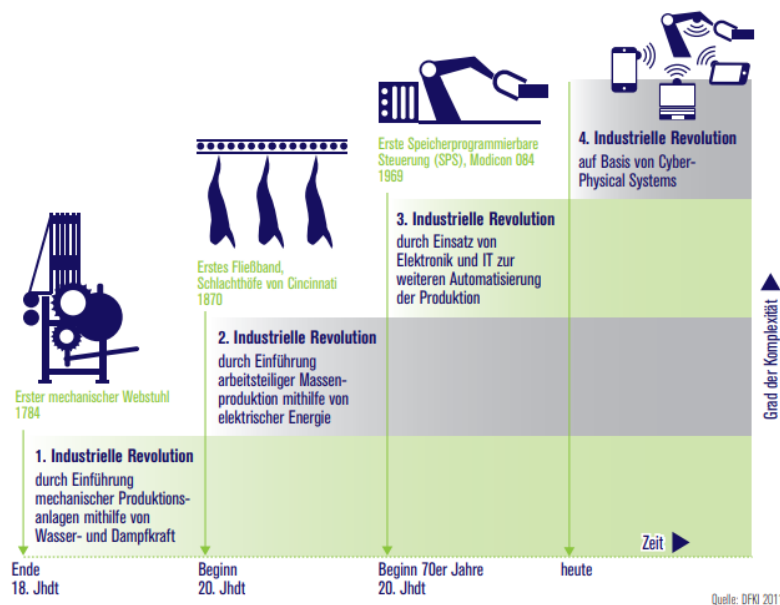


Abbildung 1: Überblick über die verschiedenen Stufen der industriellen Revolution [KaWH13 S. 17]

und zeigt ebenfalls auf, dass der Grad der Komplexität des Gesamtsystems mit jeder neuen Stufe weiter erhöht wird.

Die Abbildung beschreibt die vierte industrielle Revolution insbesondere auf Basis von Cyber-physischen Systemen (kurz CPS). Dieser ursprünglich aus Amerika stammende Begriff, beschreibt ein CPS ursprünglich als ein System, welches aus einem berechnenden Anteil (virtuelle Komponente) und einem ausführenden Anteil (physikalische Komponente) besteht.³ Da diese Definition sich kaum von einem

³ Vgl. [LeSe15 S.1].

klassischen Embedded System oder digitalen Regler unterscheiden lässt, wurde sie in Deutschland insbesondere im Rahmen der acatch Studie⁴ vom Verein Deutscher Ingenieure (VDI) wie folgt erweitert:

„Cyber-Physical Systems (CPS) sind gekennzeichnet durch eine Verknüpfung von realen (physischen) Objekten und Prozessen mit informationsverarbeitenden (virtuellen) Objekten und Prozessen über offene, teilweise globale und jederzeit miteinander verbundene Informationsnetze.“ [BeKo13, S.2]

Stützend auf dieser Definition wird Industrie 4.0 im Rahmen dieser Arbeit wie folgt definiert:

Der Begriff Industrie 4.0 umfasst, Methoden, Prozesse und Lösungen, welche auf Basis von CPS individuelle, wertsteigende Produktionsprozesse ermöglichen. Charakteristisch für Industrie 4.0 Lösungen sind die Vernetzung einzelner Automatisierungs- und Informationskomponenten untereinander. Durch diese Kombination entstehen intelligente, weitgehend selbstorganisierte autonome Fertigungssysteme, in welchen die internen relevanten Daten aufbereitet und über eine geeignete Schnittstelle nach außen zur Verfügung gestellt werden. (eigene Definition auf Grundlage von [Wasi00a] und [BaHV14]).

Aussage dieser Definition ist, dass insbesondere durch die Kombination einzelner Automatisierungskomponenten untereinander und mit einem oder mehreren Informationssystemen, wertsteigende Produktionsprozesse ermöglicht werden. Dies bedeutet nicht zwangsweise, eine steigende Produktion, sondern auch die autonome Fertigung, individueller, den Kundenwünschen angepassten, Produkten, sowie durch eine passende Datenaufbereitung und Auswertung effizientere Systeme zu schaffen.

Ein Anwendungsbeispiel für erstes ist die Smart Factory. Eine Fabrik, in welcher alle Komponenten untereinander vernetzt sind, sowie über geeignete Schnittstellen nach außen kommuniziert. Über diese Schnittstellen können die Kunden beispielsweise Bestellungen aufgeben, oder Fertigungskomponenten selbstständig fehlende Teile nachbestellen [Wasi00b].

⁴ Deutsche Akademie der Technikwissenschaft, welche sich ausführlich mit CPS Systemen auseinandersetzt.

2.1.2 Veränderte Aufgabenbereiche des Menschen im Kontext Industrie 4.0

Durch die Möglichkeiten der Smart Factory kann es insbesondere in großen Unternehmen dazu kommen, dass ganze Produktionshallen autonom fertigen und sich nur noch wenige bis gar keine Menschen in diesen befinden. Mit steigender Größe der Produktionsstrecke, steigt i.d.R. auch die Komplexität des Gesamtsystems und dementsprechend auch der Informationsfluss. Das heißt, dass eine viel größere Datenmenge anfällt.

Der Mensch selbst spielt innerhalb des Produktionsprozesses in der Smart Factory zwar keine aktive Rolle mehr, er ist aber trotzdem noch Teil des großen Ganzen. Insbesondere wenn es seine Aufgabe ist, die Prozesse zu überwachen und ggf. in diese einzugreifen, ist es wichtig, dass er einen schnellen Zugriff auf die relevanten Daten des Produktionsprozesses bekommt. Die rechtzeitige Bereitstellung der richtigen Daten wird auch unter dem Begriff „Time-to-Content“ zusammengefasst [Augm00a].

Der Augmented Operator

Im Kontext Smart Factory und Industrie 4.0 wird der Mensch oft als „Augmented Operator“ bezeichnet [Mitd00]. Der Begriff darf nicht mit Augmented Reality verwechselt werden, und der einzige Zusammenhang zwischen den beiden ist lediglich, dass AR eine mögliche Lösungstechnologie für den Augmented Operator zur Verfügung stellt (mehr dazu in den folgenden Abschnitten).

Unter Augmented Operator wird im Allgemeinen verstanden, dass der Mensch im Hintergrund über die Fertigungsprozesse wacht, das heißt diese kontrolliert und ggf. eingreift, aber auch, dass er die Abläufe optimiert.

Darüber hinaus gibt es noch weitere Aufgabenbereiche für den Menschen innerhalb der Smart Factory. Einer der fundamentalsten ist der Entwurf und die Einrichtung der Maschine (Anlagenentwurf), sowie das Erweitern bestehender Fabriken um neue Produktionslinien oder sonstigen Komponenten (Anlagenerweiterung). Überdies kann er Aufgaben im Bereich der Forschung oder Prototyping übernehmen.

Ein anderer eher klassischerer Aufgabenbereich ist die Wartung und Reparatur von Maschinen. Die erhöhte Komplexität der Systeme erhöht auch gleichzeitig die Schwierigkeit der Fehleridentifikation und Reparatur dieser. Ohne geeignete Mensch-Maschinen Schnittstellen wäre es daher kaum möglich, die Fehlerursache überhaupt zu identifizieren. Dies ist u.a. auch in anderen Bereichen wie beispielsweise der Reparatur

von KFZ zu erkennen, dort hat die Einführung der Fehlerdatenspeicher den Vorgang bei Reparaturen nachhaltig verändert [Reif16, S. 336 ff.].

Unabhängig von der konkreten Aufgabe des Menschen, habe alle Einsatzfelder eine Gemeinsamkeit. Letzten Endes benötigt der Augmented Operator eine Möglichkeit um die aktuellen Maschinenzustände zu visualisieren. Diese Möglichkeiten werden in den folgenden Abschnitten genauer erörtert.

2.2 Mensch Maschinen Schnittstellen Technologien

Möglichkeiten um Maschinenzustände zu visualisieren und ggf. mit der Maschine zu interagieren werden unter den Begriffen Mensch Maschinen Schnittstelle (kurz MMS), oder Benutzerschnittstellen zusammengefasst. Hierbei handelt es sich im einfachsten Fall um einen Schalter um die Maschine ein- und auszuschalten, sowie um eine LED, welche den aktuellen Betriebsstatus der Maschine anzeigt.

Diese eben erwähnte sehr einfache Lösung bietet sich natürlich nicht für komplexere industrielle Systeme an. Daher werden in den folgenden Abschnitten zunächst klassische MMS betrachtet und anschließend Mixed-Reality Lösungen.

2.2.1 Klassische Industrie Devices und portable Devices

Die klassische MMS Technologie ist ein fest verbauter Panel-PC, welcher i.d.R. direkt an die Fertigungslinie montiert ist. Abbildung 2 zeigt einen Panel-PC der Firma Siemens, welcher auch in ähnlicher Ausführung an der Modelfabrik der TU Clausthal verbaut ist (siehe Abschnitt 3.1).



Abbildung 2: Abbildung eines Siemens SIMATIC Panel PCs 877 [Indu00]

Die Vorteile der Panel-PCs sind, dass diese i.d.R. sehr robust gebaut sind und dementsprechend dafür ausgelegt sind, auch in robusten Produktionsumgebungen eingesetzt zu werden. Darüber hinaus sind diese genau wie SPS dazu ausgelegt, 24/7 in

Betrieb zu sein und dadurch, dass sie fest an der Anlage verbaut sind, findet der Datenaustausch sehr schnell statt.

Der große Nachteil hingegen ist die Standortbegrenzung. Dadurch eignen sich diese nicht als zentrale Lösung, sondern dienen immer nur als lokale Lösungen an der verbauten Stelle.

Eine Möglichkeit um diesem gravierenden Nachteil entgegen zu steuern, bieten portable Devices. Der Begriff umfasst alle tragbaren Lösungsmöglichkeiten, wie Laptops, Tablets und Smartphones.

Für robustere Einsatzbereiche bieten einige Firmen spezielle Tablets und Laptops an. Von den normalen unterscheiden diese sich aber auch nur darin, dass sie widerstandsfähiger sind und das OS ggf. direkt auf die Anwendung zugeschnitten wird. Die Verbindung zur Maschine erfolgt i.d.R. über WiFi.

Der größte Vorteil ist hierbei natürlich die Flexibilität, welche durch die Standortunabhängigkeit entsteht. Weitere Vorteile ergeben sich je nach Device durch die zusätzliche Sensorik. So könnte ein Smartphone beispielsweise noch zur Standortbestimmung via GPS genutzt werden, oder jedes Device, welches über eine Kamera verfügt, zur Darstellung von AR-Anwendungen (siehe Abschnitt 2.3.2).

Ein nicht zu vernachlässigender Nachteil bei portable Devices entsteht genau dann, wenn diese die Informationen für eine Vielzahl von Anlagen und Maschinen enthalten. Hierbei ist der Augmented Operator vom User Interface Design abhängig, wie lange er benötigt durch das Menü zu navigieren, bis er die gewünschten Informationen erhält. Bei der Vielzahl von Daten kann dies ggf. wichtige Zeit kosten (Time-to-Content Problem). Des Weiteren können insbesondere bei Wartungsarbeiten freie Hände erforderlich sein und es ist von Nachteil, wenn das Device mit einer oder sogar beiden Händen getragen werden muss.

Dementsprechend sind auch portable Devices für neue Anwendungsfälle ungeeignet.

2.2.2 Head Mounted Display

Head Mounted Displays (HMD) sind Projektionseinheiten, bei denen die darzustellenden virtuellen Informationen unmittelbar vor dem Auge erfolgen. Der Begriff stammt aus dem Englischen und bedeutet auf Deutsch so viel wie am Kopf befestigte Anzeige. Ein anderer Begriff hierfür im Deutschen ist Datenhelm.

Das Prinzip der HMD-Projektion wird in der folgenden Abbildung 3 dargestellt. Drei Leuchtdioden erzeugen die drei Primärfarben, Rot, Grün und Blau (RGB), welche durch ein Prisma gebrochen werden und anschließend über einen Polarisationsfilter auf ein Display (in der Abbildung ein LCoS⁵-Display) geworfen und dargestellt. Bei Virtual Retina Displays (VRD) werden die Informationen darüber hinaus noch direkt auf die Netzhaut eines Betrachters projiziert. [vgl. Date17]

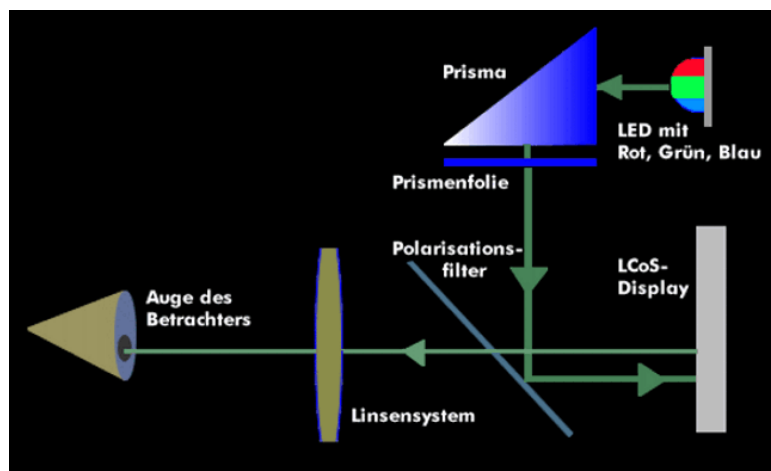


Abbildung 3: Prinzip der HMD-Projektion [Date17]

Datenhelme gibt es hierbei in unterschiedlichen Ausprägungen. In einfachsten Fall handelt es sich um eine reine Videobrille, welche die Bildinformationen einfach nur wie ein normales Display vor dem Auge wiedergibt, oder in erweiterter Form um HMDs welche beispielsweise die Kopfbewegung des Betrachters über eingebaute Sensorik auswerten. Diese Techniken, werden insbesondere im Bereich der Militär- und Medizinforschung, aber auch in der Computerspielbranche eingesetzt.

Diese Techniken werden auch unter dem Begriff Mixed Reality behandelt, welcher im folgenden Abschnitt weiter ausgeführt wird.

2.3 Mixed-Reality

Der Begriff Mixed-Reality (engl. für vermischte/gemischte Realität) wurde ursprünglich in von Paul Milgram⁶ in einem Paper 1994 eingeführt. Es beschreibt, wie in Abbildung 4 zu sehen, Umgebungen, welche die natürliche Umgebung (physical Reality) mit einer computererzeugten Umgebung (digital Reality) vermischen [MiKi94].

⁵ Liquid Crystal on Silicon bezeichnet eine Technik für Anzeigegeräte. Ähnlich zu LCD Displays.

⁶ Kanadischer Wissenschaftler und Professor für Maschinenbau- und Wirtschaftsingenieurwesen an der Universität Toronto.

Der Begriff wurde insbesondere von Microsoft in den letzten Jahren wieder verstärkt verwendet, welche diesen gemeinsam mit der Einführung der HoloLens prägten. Genau wie bei Paul Milgram umfasst er dabei das gesamte Realitäts-Virtualitäts-Kontinuum [MiKi94] mit Ausnahme der reinen Realität- oder Virtualität (physical- and digital Reality).

Realitäts-Virtualitäts-Kontinuum umfasst dabei die Gesamtheit aller vermischten Lösung von Augmented Reality (Abbildung 4 links) bis zur Virtual Reality (Abbildung 4 rechts). Beide werden in den folgenden Absätzen genauer vorgestellt.

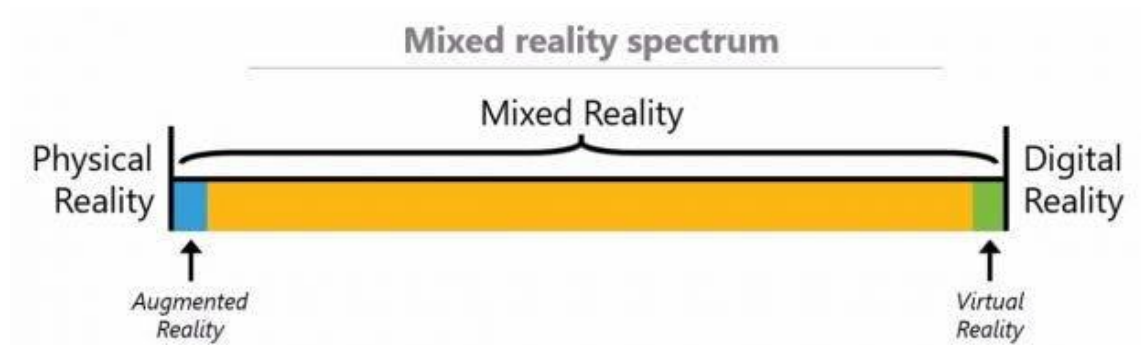


Abbildung 4: Mixed Reality Spektrum [00a]

2.3.1 Virtual Reality

Sehr nahe an der digitalen Realität befindet sich die virtuelle Realität. VR selbst ist noch ein sehr junges Wissenschaftsgebiet, weswegen es noch keine einheitliche Definition hierfür gibt [DBGJ14]. VR hat sich als aber Werkzeug in der Industrie zur Produktentwicklung bereits durchgesetzt hat und wird auch wie andere HMDs insbesondere im militärischen Bereich zu Ausbildungszwecken eingesetzt.

Eine der früheren Definition beschreibt VR als:

Virtual Reality refers to immersive, interactive, multi-sensory, viewer-centered, three-dimensional computer generated environments and the combination of technologies required to build these environments (Carolina Cruz-Neira⁷, SIGGRAPH '93 Course Notes „virtual Reality Overview“).

Eine Charakteristik der VR lässt sich am besten aufzeigen, wenn man diese im Vergleich zur traditionellen Computergrafik analysiert. Die VR Definition stützt sich explizit auf der 3D Grafik, insbesondere auf die Darstellung in Echtzeit. Zur Darstellung werden drei

⁷ Amerikanische Professorin an der University of Arkansas at Little Rock, welche sich insbesondere mit Technologien um Kontext VR beschäftigt.

dimensionale Displays (i.d.R. HMDs) verwendet und dazu kommt Sensorik, welche beispielsweise den Blickwinkel anhand von Kopfbewegungen erkennt und in die VR überträgt. Die folgende Tabelle 1 zeigt weitere Vergleichsmerkmale der VR im Vergleich zur klassischen Computergrafik:

Tabelle 1: Merkmale von VR im Vergleich zu konventioneller Computergraphik [DBGJ14 S.14 Tab. 1.1]

3D-Computergrafik	Virtuelle Realität
Rein visuelle Präsentation	Multimodale Präsentation: visuell, akustisch, haptisch
Präsentation nicht notwendigerweise zeitkritisch	Echtzeitdarstellung
Betrachterunabhängige Präsentation (exozentrische Perspektive)	Betrachterabhängige Präsentation (egozentrische Perspektive)
Statische Szene oder vorberechnete Animation	Echtzeitinteraktion und -simulation
2D-Interaktion (Maus, Tastatur)	3D-Interaktion (Körperbewegung, Hand-, Kopf- und Körpergestik) + (ggf.) Spracheingabe
Nicht-immersive ⁸ Präsentation	Immersive Präsentation

2.3.1.1 Augmented Virtuality

In einigen Darstellungen des Realitäts-Virtualitäts-Kontinuums wird neben der AR und VR oftmals noch explizit die Augmented Virtuality (AV) erwähnt. In der Abbildung 4 wäre diese links neben der VR einzuordnen. Sie beschreibt eine Symbiose aus der virtuellen Welt, mit Informationen aus der realen Welt [Av(a17)] und besteht überwiegend aus virtuellen Räumen in welchen reale Objekte und Personen eingebunden werden und in Echtzeit mit der virtuellen Welt interagieren. Ein Beispiel hierfür wäre die Einbindung von Videokonferenzteilnehmer in einen virtuellen Besprechungsraum.

AV sowie VR eignen sich zum Beispiel zur Fabrikplanung, oder für simulative Anwendungen, welche nicht auf reale bereits existierende Anlagen bezogen werden sollen. Als reines HMI hingegen sind sie komplett ungeeignet, da die Komponenten der

⁸ Immersiv, abgeleitet vom englischen Begriff „immersion“ (eintauchen auf Deutsch), beschreibt wie sehr der Nutzer in eine virtuelle Welt eintaucht bzw. als wie real er sie wahrnimmt.

physikalischen Realität nicht mehr sichtbar wären bzw. simuliert oder virtuell abgebildet werden müssten.

2.3.2 Augmented Reality

Die AR ist im Gegensatz zur VR im Realitäts-Virtualitäts-Kontinuum sehr nah an der physikalischen Welt angeordnet. Augmented Reality (engl. für erweiterte Realität) bildet im Gegensatz zur AV nicht reale Objekte auf eine VR ab, sondern virtuelle Objekte auf die reale Welt. Ein sehr bekanntes Beispiel hierfür ist das Einblenden der Abseitslinie bei Fußball Übertragungen (siehe Abbildung 5: Abseitslinie im Fußball [Vide18])



Abbildung 5: Abseitslinie im Fußball [Vide18]

Allgemein lässt sich AR also wie folgt definieren:

„Als Erweiterte Realität oder Augmented Reality (AR) wird eine computergestützte Wahrnehmung bezeichnet, bei der sich reale und virtuelle Welt vermischen. Über die gerade betrachtete reale Welt werden in Echtzeit Textinformationen und Grafiken geblendet. Die Anwendungszwecke reichen von der Information über die unmittelbare Umgebung, über die ins Sichtfeld eingeblendete Navigation bis hin zu Spielen.“ [Augm00b]

Diese Definition lässt sich sehr gut an Abbildung 6 erklären. In dieser wird die physikalische Realität (rechts) um ein einzelnes virtuelles Objekt (links) erweitert. Das Resultat, also die AR, ist in der Mitte dargestellt.



*Abbildung 6: Verschmelzung einer realen Umgebung (links) und eines virtuellen Objektes (rechts) zur AR (Mitte)
[DBGJ14 S. 242]*

AR wird bereits in den verschiedensten Anwendungsbereichen eingesetzt u.a.:

- Anzeige von Zusatzinformationen in der Konstruktion, Medizin und Wartung
- Navigation: Insbesondere auch zur transparenten Informationsdarstellung (z.B. auf der Windschutzscheibe des Autos) oder für indoor Navigation
- Videospielbereich: Insbesondere das Spiel Pokemon GO, hat einen Hype rund um AR-Mobile Games entfacht.

Für tiefere technische Details und weitere Informationen zu AR sei auf das Kapitel 4.1 Augmented Reality in der Forschung verwiesen.

2.3.2.1 Augmented Reality Hardware

Im Gegensatz zur VR muss für eine AR-Anwendung nicht zwangsweise eine Datenbrille verwendet werden. Für eine rein visuelle AR-Anwendung wird ist es lediglich notwendig, dass die Umgebung mithilfe einer Videokamera erfasst werden und möglichst in Echtzeit, inklusive der AR-Erweiterungen, dargestellt werden kann.

Insbesondere kleine AR-Anwendungen, welche zur Unterhaltung oder nur zur Darstellung kleiner virtuellen Objekte in einer realen Umgebung verwendet werden, werden oft auf Smartphones oder Tablets ausgeführt. Das vermutlich bekannteste AR-Spiel ist hierbei Pokemon GO, welches im Juli 2016 von der Firma Niantic Inc. veröffentlicht wurde. Das Spiel, welches für Android und iOS veröffentlicht wurde, bietet den Spielern die Möglichkeit, mithilfe ihres Smartphones ihre Umgebung zu erkunden und virtuelle Monster (Pokemon), welche mithilfe von AR auf ihre reale Umwelt abgebildet werden, einzufangen (siehe Abbildung 7). Im Gegensatz zu anderen AR-

Anwendungen wird der Videostream hier lediglich zur Darstellung verwendet. Das heißt, dass das AR-Objekt (in diesem Fall das Pokemon) nicht korrekt in seine Umgebung eingepasst wird, sondern unabhängig von den Informationen des Videostreams nur darauf projiziert, wie ebenfalls in Abbildung 7 deutlich zu erkennen ist.



Abbildung 7: Ausschnitt aus dem AR-Spiel Pokemon GO

Ein weiteres Merkmal der AR dieses Spiels ist, dass neben den visuellen Informationen noch der Standort des Spielers verwendet wird, welcher via GPS ermittelt wird.

Der entstehende Trend von AR in mobile Applikationen ging auch an der großen Hard- und Software Herstellern nicht vorbei, so dass insbesondere Google (mit dem ARCore, ehemals Projekt Tango) und Apple (dem AR-Kit) in den letzten Jahren ihre eigenen AR-Frameworks veröffentlichten. Daneben existieren noch eine Vielzahl weiterer AR-Frameworks, wie Wikitude, ARMedia und das ARToolkit. Letzteres ist hierbei das älteste und längste am Markt erhältliche Framework, welches bereits 1999 erschienen ist und 2015 als Open Source Release erschien.⁹

Da nicht jedes Framework für jede Plattform (IOS/Android/UWP) verfügbar ist und im Rahmen dieser Arbeit insbesondere die HoloLens von Microsoft verwendet werden soll, wird im weiteren Verlauf dieser Arbeit insbesondere das Vuforia Framework weiter betrachtet und genauer beschrieben in Kapitel 6.2.4 .

⁹ Siehe auch <https://github.com/artoolkit>

Eine weitere potentielle Einsatzplattform für AR-Anwendungen sind die bereits eingeführten Datenbrillen. Im Gegensatz zu VR-Brillen, sind AR-Displays transparent. Das heißt im Gegensatz zu Smartphones, welche visuelle AR-Anwendungen darstellen, indem sie ihre Umwelt erst aufnehmen, anschließend um die AR-Informationen erweitern und zum Schluss diese auf einen Videostream abbilden, werden in AR-Datenbrillen, die AR-Informationen direkt auf die Realität abgebildet. Eine Aufnahme der Umwelt mithilfe einer Kamera, wäre also nicht mehr zwangsläufig erforderlich (wird aber i.d.R. trotzdem angewendet um die Objekte besser in die reale Umwelt zu positionieren).

Insofern lediglich Informationen visualisiert werden, aber nicht auf die Informationen der Umwelt (die vom Auge wahrgenommene Szene) zurückgreifen, werden die Geräte als Head-Up-Displays (HUD) bezeichnet, ansonsten als EyeTaps. HUDs werden im Folgenden nicht weiter betrachtet, da sie als Mixed-Reality Anwendungen nicht in Frage kommen.

Der Begriff EyeTap ist gleichzeitig auch der Name eines Unternehmens, welches vom EyeTap-Erfinder Steve Mann¹⁰ gegründet wurde [Eyet00]. Eines der bekanntesten EyeTap Devices ist das Google Glass (siehe Abbildung 8). Die Google Glass, verfügte neben einer Kamera, zur Aufnahme ihrer Umwelt, über diverse weitere Sensorik, womit u.a. auch die Blickrichtung und Kopfposition ermittelt werden konnte. Das Bild wurde über einen LCoS-Microdisplay erzeugt und über ein Glasprisma in das Blickfeld des Benutzers projiziert (siehe auch Abbildung 3). Die Google Glass hat zwar eine mediale Welle geschlagen, insbesondere wurde eine riesige Datenschutzdebatte entfacht, und teilweise auch als technischer Meilenstein gefeiert [Goog00], schaffte es aber trotz allem letztlich nie in den freien Verkauf und das Projekt wurde von Google eingestellt.



Abbildung 8: Google Glass Explorer Edition (2014)

¹⁰ Kanadischer Informatik Professor an der University of Toronto. Sein Forschungsschwerpunkt ist insbesondere AR.

Der Vorteil von Eyetaps gegenüber Smartphones/Tablets als AR-Device ist, dass die Steuerung bzw. der Blickwinkel des Devices immer dieselbe wie die Blickposition des Kopfes ist. Der Aufwand das Gerät händisch zu positionieren entfällt und ein weiterer positiver Nebeneffekt, insbesondere für industrielle MR-Anwendungen ist, dass der Benutzer dadurch freie Hände hat. Darüber hinaus verfügen diese oft noch über zusätzliche Sensorik, beispielsweise zur besseren Wahrnehmung der Umwelt. Eine Datenbrille welche insbesondere für Mixed-Reality-Anwendungen ausgelegt ist, da sie ihre Umwelt nicht nur über den Videostream sondern noch über zusätzliche Sensorik erfassen ist die HoloLens von Microsoft. Diese wird im Folgenden genauer vorgestellt.

2.3.2.2 Microsoft HoloLens

Nachdem es nach Einstellung der Google Glass wieder etwas ruhiger um AR-HMDs geworden ist, stellte Microsoft Anfang 2015 auf einer Präsentation die HoloLens vor. Im Gegensatz zum „klassischen“ EyeTap und der Google Glass verfügt die HoloLens über Displays vor beiden Augen (siehe Abbildung 9) sowie über eine Vielzahl weiterer Sensorik zur Umfeld Erfassung. Mithilfe dieser ist es der HoloLens möglich sich intern ein eigenes Abbild seiner Umwelt zu erstellen und darüber hinaus auch seine eigene Position in diesem Abbild zu ermitteln (mehr dazu siehe Simultaneous Localization and Mapping).

Eine weitere Besonderheit, welche derzeit noch einzigartig im Vergleich zu den Mitbewerben ist, ist die Architektur. Neben den klassischen Komponenten wie der CPU, GPU und RAM befindet sich in der HoloLens noch eine Holographic Processing Unit (HPU). Dieser Chip ist insbesondere im Vergleich zu herkömmlichen CPUs darauf ausgelegt, ständige und in großen Massen eingehende digitale Signale zu verarbeiten, welche die Sensoren und die verschiedenen HoloLens Kameras liefern und dient nur dazu die Projektionen zu berechnen und richtig im realen Raum zu platzieren [Kolo16].



Abbildung 9: Abbildung der Microsoft HoloLens [Mico00]

Des Weiteren bildet die HoloLens alleine schon einen völlig funktionsfähigen Computer und muss nicht wie die meisten VR-Brillen in Kombination mit einem weiteren PC betrieben werden. Zusätzliche Hardware für User Eingaben sind zwar optional erhältlich, aber auch nicht zwingend notwendig, da die HoloLens auf verschiedene Gesteneingaben reagiert.

Die HoloLens wurde bisher noch nicht als Version für Privatanwender veröffentlicht, kann aber von jedem als Developer Edition bezogen werden. Von der Größe und dem Tragekomfort ist die derzeitige Version definitiv auch eher noch eine Entwicklerversion, es gilt aber zu erwarten, dass Microsoft dort zukünftig nachbessert und in den kommenden Jahren auch eine Version, welche insbesondere auf Privatkunden abzielt veröffentlichen wird. Überdies wird die zweite Version der HoloLens über eine verbesserte HPU verfügen, welche insbesondere auch eine Recheneinheit für künstliche Intelligenz enthält [Seco17].

2.3.3 Zusammenfassung: Microsofts Mixed-Reality

Wie einleitend bereits erwähnt, etablierte Microsoft gemeinsam mit der HoloLens auch erneut das Schlagwort Mixed-Reality. Mixed-Reality beschreibt hierbei nicht nur einerseits das gesamte Realitäts-Virtualitäts-Kontinuum, sondern fokussiert insbesondere auch die Interaktion zwischen der realen und der physikalischen Welt.

Da die Definition von MR, VR und AR in der Literatur nicht eindeutig sind, die Interaktion zwischen der physikalischen und der realen Welt aber im Rahmen dieser Arbeit einen besonderen Stellenwert erhält, wird die Definition aus der Sicht von Microsoft hier noch einmal zusammengefasst und auch im Rahmen dieser Arbeit für Mixed-Reality Anwendungen verwendet.



Abbildung 10: Unterschiede der drei Technologien aus der Sicht von Microsoft: In der AR ist das Bild im Vergleich nur MR nicht korrekt in die Umgebung eingepasst [Bare17]

Am besten lässt sich diese Definition noch einmal an der Abbildung 10 erklären. Ganz Links wird dort ein virtuelles Objekt in einer VR gezeigt. In einer AR hingegen wird das virtuelle Objekt auf die reale Welt abgebildet. Genau wie in dem AR-Spiel Pokemon GO (siehe auch Abbildung 7) wird das Objekt zunächst einfach nur auf die Realität abgebildet. In der Mixed-Reality hingegen interagiert, das Objekt aber mit der realen Umwelt, das heißt, dass das in Abbildung 10 gezeigte Sofa, für das virtuelle Objekt ebenso wie für den Menschen eine natürliche Barriere darstellt und es sich deswegen nur dahinter oder davor befinden kann. Überdies steht das Objekt in der MR korrekt auf dem Boden und schwebt nicht mehr wie der AR in der Luft, die Schwerkraft ist also in der MR genauso wie in der physikalischen Welt aktiv. Die Unterschiede werden auch noch einmal in Tabelle 2 zusammengefasst.

Tabelle 2: Unterschiede der drei Technologien aus der Sicht von Microsoft [Wasi17]

	VR	AR	MR
Reale Welt vorhanden	Nein	Ja	ja
Interaktion mit realer und virtueller Welt?	Nein	ja	ja
Interaktion zwischen Inhalten realer und virtueller Welt?	Nein	Nein	ja

2.3.4 Einsatzmöglichkeiten für Mixed Reality im industriellen Kontext

Für MR-Anwendungen gibt es ein breites Spektrum von Anwendungsgebieten. Laut einer Deloitte Studie werden Unternehmen im Jahr 2020 knapp 850 Millionen Euro in

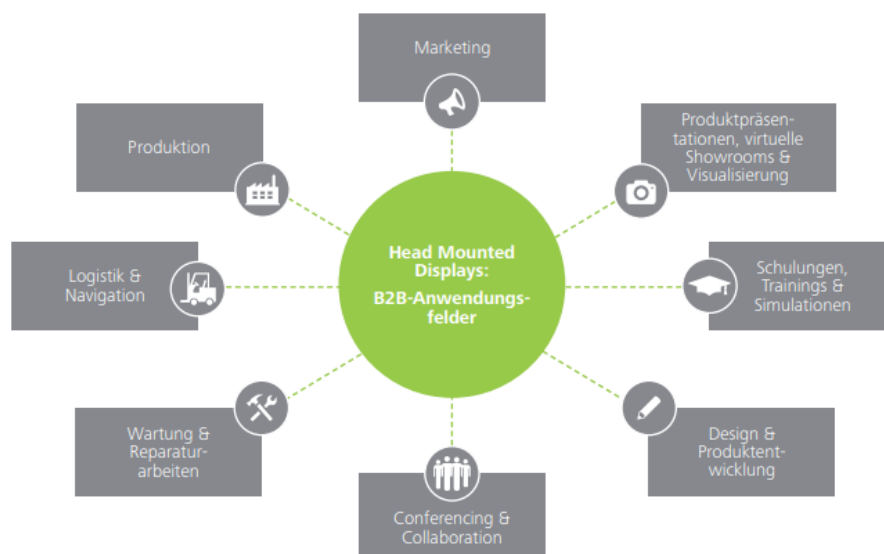


Abbildung 11: B2B Anwendungsfelder für HMDs [EsOL16]

entsprechende Lösungen investieren. Der Großteil wird nicht in die Hardware, sondern in innovative Anwendungen fließen [EsOL16, S.3].

Abbildung 11 gibt hierbei einen Überblick über die potentiellen Anwendungsfelder für Datenhelme im Allgemeinen. Insbesondere die folgenden Bereiche sind dabei für MR-Anwendungen interessant:

- Design & Produktentwicklung: Insofern, dass das Produkt im realen Umfeld betrachtet werden soll oder sogar in Verbindung zu diesem steht.
- Logistik & Navigation: Insbesondere im Bereich der Indoor-Navigation bietet beispielsweise die HoloLens ein hohes Potential.

Weitere Anwendungsfelder wie Wartung & Reparatur, sowie Produktion und Simulation werden im Folgenden noch etwas detaillierter dargestellt, da diese Arbeit die Grundlagen für diese schaffen soll.

2.3.4.1 Wartung und Reparatur

Ein besonderes Potential bietet sich im Bereich der Wartung und Reparatur von Maschinen. Zunächst alleine dadurch, dass der Mitarbeiter durch das Tragen eines HMDs freie Hände hat. Darüber hinaus können die Möglichkeiten des internen Mappings dazu genutzt werden, dass der Mitarbeiter bei Fehlerfällen direkt zum Ursprung des Fehlers navigiert wird und dass dort bereits Handlungsempfehlungen gegeben werden. Durch die Möglichkeiten der Objektidentifizierung, können diese dann direkt an den entsprechenden Anlagenkomponenten angezeigt werden.

In Kombination mit Deep Learning Verfahren könnten darüber hinaus noch potentielle Industrie 4.0 Lösungen entstehen, welche durch das Auftreten von Fehlern, diese Analysieren und bei ähnlichen Fehlern Handlungsempfehlungen aussprechen, auch bekannt unter dem Stichwort Smart Maintenance.¹¹

2.3.4.2 Visualisierung des Produktionszustandes

Bei der Überwachung bzw. Kontrolle von Produktionsprozessen bietet sich ein weiteres potentielles Einsatzfeld. Der Augmented Operator erhält durch AR-Anwendungen direkt die Möglichkeit, die internen Maschinenzustände mit den realen zu vergleichen (Soll/Ist-Vergleich). Als kleines Beispiel sei hier ein Hochregallager genannt, in welchem die

¹¹ Smart Maintenance steht für intelligente, lernorientierte Instandhaltung. Siehe auch [Inst16, S.9 ff].

Stückzahl nicht über Sensorik erfasst werden kann, sondern nur geändert wird, wenn maschinell etwas ein- oder ausgelagert wird. Hierbei könnte es vorkommen, dass Teile durch einen Mitarbeiter entfernt, oder hinzugefügt werden, dies aber nicht ins System eingepflegt wird. Eine MR-Anwendung kann die Stückzahl des aktuellen Maschinenzustands direkt auf dem dazugehörigen Hochregallager visualisieren, so dass der Abgleich für den Mitarbeiter sehr schnell und fehlerfrei von statten geht.

Dieses Beispiel lässt sich verallgemeinert auf fast alle Produktionsprozesse abbilden bei denen der Mensch eine Kontroll- bzw. Überwachungsfunktion ausführt. Des Weiteren lassen sich hiermit stichprobenartige Kontrollen (z. B. Werkstücke im Umlauf) oder Qualitätskontrollen ausüben (hierbei könnte die MR-Anwendung z. B. selbstständig die Maße des Werkstückes analysieren und für den Kontrolleur aufbereiten).

Die Ergebnisse dieser Arbeit sollen später auch in dem Forschungsprojekt „Generative Fertigung im Bauwesen“ umgesetzt, in welchem ein 3D Druck Verfahren mit Beton erforscht wird. Mithilfe der HoloLens sollen die aktuellen Statuswerte des Roboters und des Betondurchflusses an diesem visualisiert werden. Dabei gilt es auch als Herausforderung dem Roboter im Raum zu tracken, eine mögliche Umsetzung zeigt die Abbildung 12.



Abbildung 12: aktiver Spritzprozess im Projekt generative Fertigung

2.3.4.3 Simulation

Ein anderes etwas spezifisches Einsatzfeld ergibt sich im Bereich der MR-Simulationen. Mithilfe der HoloLens kann ein simulierter Prozess auf bereits vorhandene Komponenten abgebildet werden, oder es können Informationen an Werkstücken visualisiert werden,

welche aus einer Simulation stammen. Ein Beispiel anhand der Modelfabrik der TU Clausthal wird in dem folgenden Kapitel 3.1 eingeführt. Hierbei sollen Informationen welche aus einer Simulation stammen direkt auf reale Objekte abgebildet werden.

Ein anderes mögliches Einsatzszenario wäre den Abkühlprozess von Werkstücken welche aus einem Hochtemperaturoffen kommen zu beobachten. Der Abkühlprozess könnte dabei z. B. in Matlab/Simulink materialabhängig berechnet werden und auf das Objekt abgebildet werden (eine AR in welcher die Sinne des Menschen um die Sichtung einer Temperatur erweitert wird) und abschließend wird mit diesen Informationen die Anlage entworfen.

Das Projekt „Generative Fertigung im Bauwesen“ bietet auch für diesen Kontext ein interessantes Anwendungsfeld, da dort der Auftragsprozess mithilfe der HoloLens modelliert werden könnte und somit die Bahnplanung vorrauschauend evaluiert werden kann.

2.3.4.4 Mixed Reality Tischkicker Auswertung

Ein weiteres potenzielles Anwendungsszenario für eine MR-Anwendung wurde bereits häufig im SSE diskutiert und behandelt die Kopplung einer MR-Brille mit einem Tischkicker. Durch visuelle Trackingverfahren könnte beispielsweise dem Spielball gefolgt werden und durch intelligente Hinweise der HoloLens und aus einer Analyse des Spielflusses könnten Tipps für weitere Spielzüge an den Endbenutzer weitergegeben werden.

Darüber hinaus könnte mithilfe der HoloLens eine automatische Tor-Statistik erstellt werden und ggf. Animationen eingeblendet werden, welche besonders gute Tore herausstellen (zum Beispiel ein Feuerball bei einem Torwarttor).

3 Problemstellung

Der folgende Abschnitt beschäftigt sich mit dem konzeptionellen Problem, dass sich die virtuelle und die reale Welt aufgrund der Definition der Mixed Reality nicht mehr getrennt voneinander betrachten lassen, sondern miteinander in Beziehung stehen. Hierfür soll zu Beginn zunächst das Problem noch weiter spezifiziert und analysiert werden und resultierend aus dieser Analyse in 3 Bereiche (bzw. im folgenden Welten) zerlegt werden und anhand dieser diskutiert werden. Das Problem soll beispielhaft anhand eines simulierten Lackierprozesses an der Modelfabrik der TU Clausthal erläutert werden.

3.1 Ausgangssituation des Beispiels

Bei der eben eingeführten Modelfabrik handelt es sich um ein verkleinertes Abbild einer realen Fabrik (siehe Abbildung 13), welche aus einem Hochregallager und verschiedenen Förderbändern besteht. Darüber hinaus setzt sie sich aus verschiedenen Aktoren, wie einem kompressionsgesteuerten Greifer für das Hochregal und unterschiedlicher Sensorik zusammen, u.a. Lichtschranken. Das ursprüngliche Programm der Modelfabrik lässt simultane zylinderförmige Werkstücke über die Förderbänder laufen, welche durch einen integrierten RFID¹² Chip eine eindeutige Identität besitzen. Die RFID Daten können mit den unterschiedlichen RFID Köpfen sowohl gelesen als auch verändert werden.

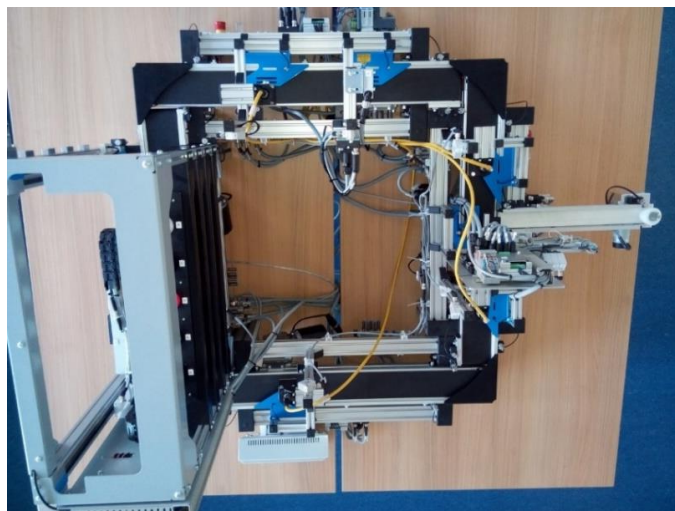


Abbildung 13: Bild der Modelfabrik der TU Clausthal

¹² RFID steht für die englische Bezeichnung „radio-frequency identification“ und bezeichnet eine Technik zum Datenaustausch über Radiowellen auf kurzen Distanzen.

Ein potentielles Industrie 4.0 Szenario für die Modelfabrik ist die individuelle Fertigung. Der Kunde könnte beispielsweise sein Produkt individuell gestalten, z.B. Farbe, Ausprägung etc. und die Kundeninformationen werden an den Werkstücken gemeinsam mit der Bestellnummer visualisiert. Damit erhält der Augmented Operator weiterhin den Überblick über alle Bestellungen. Die Informationen werden dann nach dem ein Werkstück ausgelagert wurde auf den RFID Chip eines Werkstückes geschrieben und können durch diesen auch wieder ausgelesen werden.

Die praktischen Vorarbeiten für dieses Szenario wurden bereits in vorherigen Arbeiten umgesetzt, aber bisher konnten die RFID Informationen lediglich auf dem Panel-PC der Anlage angezeigt werden. Hierfür konnte immer nur ein Werkstück ausgelesen werden und der Augmented Operator ist Standortbeschränkt.

3.2 Übersicht

Bei der praktischen Umsetzung dieser Visualisierungsmöglichkeit stößt man jedoch auf einige Probleme, da zunächst ein Konzept entwickelt werden muss, um die visuellen Informationen an der richtigen Stelle zu positionieren, sowie die richtigen Informationen (in diesem Fall der richtige Farbcode des Werkstückes) zu visualisieren.

Zur besseren Beschreibung dieses Problems sei noch einmal auf Abbildung 4 verwiesen. Genau wie dort bietet es sich an zunächst zwischen der physikalischen Welt (in der Abbildung Physical Reality) und der digitalen Welt (in der Abbildung digital Reality) zu unterscheiden.

Vereinfacht gesagt enthält die physikalische Welt hierbei alle realen Objekte die in irgendeiner Form visuell erfassbar sind und die digitale Welt alle virtuellen Datenobjekte, welche in elektronischer Form in verschiedensten Geräten vorhanden sind. Ziel dieser Arbeit soll es sein, die digitalen Informationen auf ihren dazugehörigen physikalischen Objekten mithilfe einer Mixed Reality Technologie abzubilden. Für das Beispiel bedeutet dies, dass die physikalische Welt die Modelfabrik und die dazugehörigen Werkstücke beschreibt und die digitale die Informationen auf den RFID-Tags, welche die Farbcodes enthalten, welche mit der MR dargestellt werden soll.

Die Abbildung 14 stellt hierbei nochmal die 3 unterschiedlichen Welten welche miteinander in Beziehung gebracht werden müssen. Diese sollen in den folgenden Unterpunkten genauer erläutert werden.

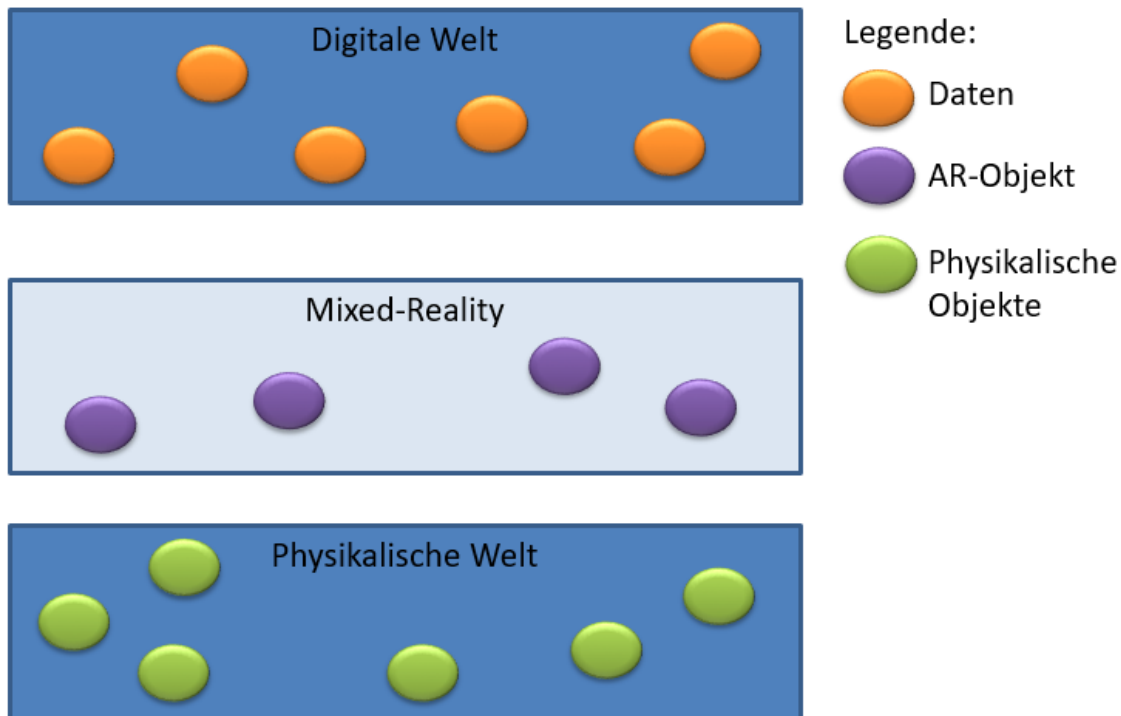


Abbildung 14: Übersicht über die unterschiedlichen Welten

3.3 Physikalische Welt

Die unterste Ebene der Abbildung 14 beschreibt die physikalische Welt. Hierbei handelt es sich um die reale Welt, in welcher wir uns Tag für Tag bewegen. Die Bestandteile dieser Welt sind physikalische Objekte. Die Idee, die Welt in Objekte einzuteilen ist hierbei nicht neu, sondern Bestandteil der Objektorientierung. Hierbei handelt es sich um eine Methodik der Informatik, welche ein System als eine Vielzahl von Objekten beschreibt, welche miteinander über Nachrichten kommunizieren [00b].

Ein Objekt beschreibt hierbei eine Sache oder einen Gegenstand, auf den man sein Interesse richtet. Für das Beispiel der Modelfabrik sind dies insbesondere die einzelnen Komponenten dieser (Förderbänder, Hochregallager, Sensoren), sowie die einzelnen Werkstücke.

Dabei soll im Rahmen dieser Arbeit nur auf sichtbare Objekte eingegangen werden, welche durch visuelle Sensoren (wie Kameras) erfasst werden können. D.H mit anderen Worten, visuell nicht erfassbare Objekte wie beispielsweise Gase, könnten zwar für andere industrielle Mixed Reality Anwendungen von Interesse sein, werden hier aber nicht weiter betrachtet.

Anhand des eingeführten Beispiels sollen nun die relevantesten Eigenschaften von physikalischen Objekten herausgearbeitet werden. Bei der Modelfabrik, welche die Werkstücke bearbeitet, ist die Maschine selbst ein statisches Objekt mit einer festen Position, wohingegen die Werkstücke dynamisch sind und ihre aktuelle Position ständig verändern. Dementsprechend ist zum einen die Position eine wichtige Eigenschaft der physikalischen Objekte und dazugehörend auch, ob es sich um eine feste oder eine veränderliche Position handelt. Des Weiteren lassen sich die Werkstücke und die Maschinenelemente gut typisieren bzw. zu einem Oberbegriff zuordnen. Die Werkstücke könnten beispielsweise anhand ihrer Form (Zylinder, Quader etc.) typisiert werden.

Zusammenfassend ergeben sich die folgenden Eigenschaften für physikalische Objekte:

- Position: Die Position beschreibt die Koordinaten des Objektes in einem vom Anwender gewählten Koordinatensystem¹³.
 - Mobilität: Hierbei wird zwischen Statisch und Dynamisch unterschieden. Ein statisches Objekt besitzt eine konstante Position, während dynamische Objekte ihre Position ständig verändern, oder zumindest über die Möglichkeit verfügen diese zu verändern.
- Typ: Jedes Objekt lässt sich einer bestimmten Gruppe zuordnen.

3.4 Digitale Welt

Die digitale Welt beschreibt die virtuelle Welt. Hier befinden sich alle für den Benutzer alle digitalen Informationen, welche nicht aufbereitet zunächst erst einmal wenig hilfreich für den Endnutzer sind. Die Daten innerhalb dieser Welt können aus den unterschiedlichsten Datenquellen stammen. Für das Beispiel mit der Modelfabrik handelt es sich hierbei unter anderem um:

- Sensordaten: Zustände der Lichtschranken¹⁴ und der RFID Köpfe
- Maschinenzustände: Geben Auskunft über den aktuellen Status der Fabrik und der Steuerungen, zum Beispiel ob ein Förderband läuft oder ob Fehler in der SPS vorliegen.

¹³ Mathematisches Hilfsmittel zur Beschreibung der Position eines Objektes. Ein bekanntes Koordinatensystem ist beispielsweise das geographische Koordinatensystem der Erde, welches einen Punkt auf der Erde mit den Längen- und Breitengraden beschreibt.

¹⁴ Hierbei handelt es sich um einen Lasersensor, welcher erkennt ob sich ein Werkstück in der Lichtschranke befindet oder nicht

- Simulationsdaten: Daten welche aus einem Simulationstool stammen, in diesem Fall die Farbe der Werkstücke.
- ERP-System Daten: Daten aus einem übergeordneten Enterprise System, zum Beispiel aus einem Bestellsystem.

Neben den bereits aufgezählten Daten existieren in der digitalen Welt noch eine Vielzahl weiterer Informationen, welche aber keinen direkten Nutzen für den Endanwender haben, (z. B. der Programmcode) und daher für den Endanwender nicht von Bedeutung sind.

Die Herausforderungen welche hierbei entstehen, sind einerseits die korrekten Daten zu selektieren und andererseits diese rechtzeitig bereitzustellen (Time-to-Content ¹⁵).

3.5 Mixed Reality Welt

Die Mixed Reality Welt befindet sich zwischen der physikalischen und der digitalen Welt. Während die beiden anderen Welten getrennt voneinander existieren können, kann die Mixed Reality Welt weder ohne die physikalische noch ohne die digitale Welt existieren. Die MR-Welt besteht aus AR-Objekten, welche dazu dienen sollen, die Informationen der digitalen Welt auf die reale Welt abzubilden. Für das eingeführte Beispiel sollen die Simulationsdaten (die Farbe des Werkstückes) auf das dazugehörige Werkstück abgebildet werden.

Ein AR-Objekt besitzt hierbei zwei wesentliche Eigenschaften:

- Aussehen: Beschreibt die Visualisierungsform des Objektes
- Position: die Position innerhalb der MR-Welt, welche in Beziehung zur realen Welt steht

Damit die AR-Objekte im Raum positioniert werden können, muss die Mixed Reality selbst ein eigenes Abbild der physikalischen Welt enthalten und seine eigene Position innerhalb dieser Welt bestimmen können. Dieses Problem ist auch als SLAM Problem bekannt und soll in dem folgenden Unterpunkt genauer erläutert werden

3.5.1 Simultaneous Localization and Mapping

Ursprünglich stammt das Simultaneous Localization and Mapping Problem (englisch für Simultane Lokalisierung und Kartenerstellung) aus dem Bereich der Robotik und beschreibt das Problem, dass ein mobiler autonomer Roboter zur Orientierung einerseits

¹⁵ Unter Time-to-Content wird die Zeitdauer verstanden, welche benötigt wird, um dem Endanwender die von ihm gewünschte Information anzuzeigen [MeRe11].

eine Karte seiner Umwelt erstellen muss und andererseits seine eigene Position innerhalb dieser Karte ermitteln muss [Pfin13].

Dieses Problem lässt sich perfekt auf Mixed Reality Anwendungen übertragen, da die Kenntnisse des Raumes auch hier essenziell sind um die MR-Objekte an den passenden physikalischen Objekten zu platzieren.

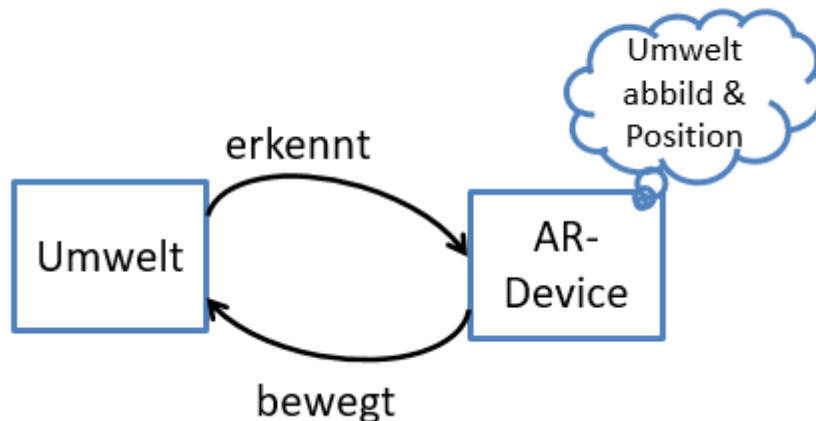


Abbildung 15: Skizze des SLAM Problems (inspiriert von [Pfin13])

Abbildung 15 skizziert noch einmal zusammenfassend diese Problematik. Das AR-Device bewegt sich in Ausschnitten der physikalischen Welt (hier als Umwelt bezeichnet) und muss diese erkennen. Durch die Erkennung legt es ein internes Abbild der Umwelt an und bestimmt seine eigene Position innerhalb dieses Abbildes. Des Weiteren muss das AR-Device später die Position der erzeugten AR-Objekte innerhalb seiner Umwelt speichern.

3.6 Zusammenfassung

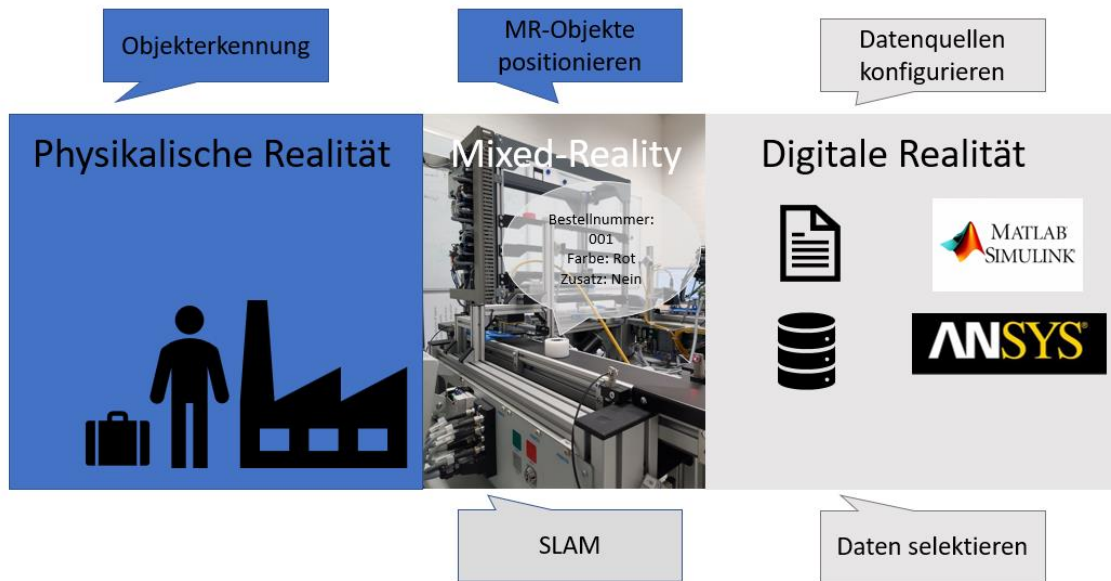


Abbildung 16: Übersicht über die Probleme innerhalb der verschiedenen Welten. Die blauen Probleme werden im Rahmen dieser Arbeit behandelt

Die obenstehende Abbildung skizziert zusammenfassend wie die drei Welten miteinander in Beziehung gebracht werden sollen. Die MR, dient dabei als Linker zwischen der digitalen und der physikalischen Welt. Die AR-Objekte bieten dabei eine Schnittstelle der Daten zu ihren dazugehörigen physikalischen Objekten.

Um einen logischen Zusammenhang zwischen diesen herzustellen muss auf jeder Betrachtungsebene (jeder Welt) eine andere Herausforderung gelöst:

- Digitalen Welt: Hier müssen die enthaltenen Daten logisch selektiert werden und ihren dazugehörigen physikalischen Objekten zugeordnet werden.
- Physikalische Welt: Es muss eine Beschreibungsmöglichkeit gefunden werden um die AR-Objekte auf ihren dazugehörigen physikalischen Objekten darzustellen. Eine besondere Herausforderung bilden hierbei dynamische Objekte, welche keine konstante Position haben. Es muss ein Verfahren gefunden werden, welches auch diese Objekte eindeutig identifiziert.
- Mixed Reality: Für die Existenz der MR muss zunächst ein Abbild der physikalischen Welt erstellt werden und anschließend muss eine Möglichkeit gefunden werden die AR-Objekte innerhalb dieses Abbildes so zu positionieren, dass sie sich die Position mit der Position der physikalischen Objekte deckt.

Hierfür muss das AR-Device ebenfalls seine eigene Position innerhalb des Abbildes bestimmen können (siehe Simultaneous Localization and Mapping)

Ausgehend von den drei Herausforderungen, soll in diesem Abschnitt noch einmal verdeutlicht werden, welche hiervon im Rahmen dieser Arbeit primär betrachtet werden soll.

Die Lösung des SLAM Problems ist abhängig von der verwendeten Hardware und integraler Bestandteil dieser. Da die HoloLens als verwendete Hardware ausgewählt wurde und diese hierfür bereits mit dem Spatial Mapping eine Lösung bereitstellt, wird dies hier nicht weiter betrachtet.

Die Herausforderung der digitalen Welt soll zwar betrachtet werden und wird für das Lösungsframework berücksichtigt, aber auch dies soll nicht der Fokus der Arbeit sein.

Primär wird das Problem der Identifikation der Objekte in der physikalischen Welt betrachtet. Hierfür soll eine Unterscheidung zwischen dynamischen und statischen physikalischen Objekten getroffen werden und für beide soll abschließend eine Möglichkeit zur Verfügung stehen, die zugehörigen digitalen Informationen an ihnen zu positionieren und zu visualisieren.

4 State of the Art

Im vorherigen Kapitel wurden die Probleme beim Entwurf einer Mixed-Reality Anwendung identifiziert und als Kernproblem die Identifikation von realen Objekten herausgearbeitet. Da dieses im weiteren Verlauf der Arbeit gelöst und mithilfe einer Augmented-Reality Anwendung visualisiert werden soll, wird in diesem Kapitel der aktuelle Stand der Forschung zum Thema Augmented Reality, sowie verschiedene Verfahren zur Objekterkennung behandelt. Des Weiteren wird im Abschluss noch das Thema Informationsfusion behandelt, da die gewählte Lösung im Rahmen dieser Arbeit ebenfalls auf diesem Ansatz basiert.

4.1 Augmented Reality in der Forschung

In der Literatur findet sich eine Vielzahl von Definitionen für das Thema AR. Erstmals wurde das Thema in den 60er Jahren [Suth65] von Ivan Sutherland¹⁶ erwähnt, welcher auch den ersten HMD: „*The Sword Of Damocles*“ entwarf.

Die meisten Definitionen die sich in der Wissenschaft durchgesetzt haben basieren auf einer Veröffentlichung von Ronald Azuma, welcher ein AR-System an den folgenden drei Charakteristiken definiert [Azum97]:

1. Kombiniert Realität und Virtualität
2. Interagiert in Echtzeit
3. Registriert 3D-Objekte

Als Nachteil dieser Charakteristik wird häufig aufgeführt, dass die Definition sich nur auf die technischen Merkmale stützt und nicht das Gesamtspektrum der AR wiedergibt, da AR auch als eine Erweiterung der Sinneswahrnehmung des Menschen definiert werden kann. Dies bedeutet, dass dem Menschen mithilfe von Sensorik Umgebungseigenschaften wie Schall, Radar, Infrarot etc., wahrnehmbar gemacht werden, welche er ohne technische Hilfsmittel nicht wahrnehmen könnte.

Da sich die Arbeit insbesondere auf das Registrieren und Identifizieren von physikalischen Objekten beschränkt, wird die folgende Definition von AR für den weiteren Verlauf dieser Arbeit gewählt:

¹⁶ Amerikanischer Wissenschaftler im Bereich der Computergrafik. Er entwickelte eine der ersten interaktiven Grafikanwendungen und erhielt 1988 den Turing Award für seine Leistungen im Bereich der Computergrafik. [Ivan00].

„Augmentierte Realität ist eine (unmittelbare, interaktive und echtzeitfähige) Erweiterung der Wahrnehmung der realen Umgebung um virtuelle Inhalte (für beliebige Sinne), welche sich in ihrer Ausprägung und Anmutung soweit wie möglich an der Realität orientieren, so dass im Extremfall (so das gewollt ist) eine Unterscheidung zwischen realen und virtuellen (Sinnes-) Eindrücken nicht mehr möglich ist.“ [DBGJ14 S. 246]

Eine visuelle AR Anwendung lässt sich daher in die folgenden 5 Schritte aufteilen:
[DBGJ14 S. 243]

1. Videoaufnahme: Ein Videostream der Umgebung des Betrachters wird aufgenommen.
2. Tracking: Die Position (Lage und Orientierung) des Betrachters wird geschätzt bzw. bestimmt. Hierbei handelt es sich um die Lösung des in Kapitel 3.5.1 behandelten SLAM-Problems.
3. Geometrische Registrierung: Hierunter wird die korrekte Einpassung der virtuellen Objekte in die Realität verstanden. Die physikalische Welt bzw. die Umgebung in welcher sich das Device befindet wird nachdem sie aufgenommen wurde registriert. Beispielsweise wird eine Wand als solche erkannt und es ist einem AR-Objekt nicht möglich diese zu durchschreiten.
4. Darstellung: Positionierung des AR Objektes in die erkannte Realität. Diese erfolgt in Abhängigkeit der aktuellen Position des Devices
5. Ausgabe: Der fertige AR-Stream wird für den Endnutzer wieder ausgegeben.

Das Problem, dass sich bei dieser Unterteilung ergibt, ist dass es keinen Punkt für das abschließende Identifizieren der physikalischen Objekte gibt. Die Definition ist vor allem darauf aus, die AR Objekte richtig zu positionieren. Da aber gerade die Identität der physikalischen Objekte in industriellen Anwendungen von großer Bedeutung ist, z.B. um Simulationsergebnisse, oder sonstige zu diesem Objekt gehörenden Objekte an diesem anzuzeigen, beschäftigt sich der folgende Abschnitt mit dem Thema der Objekterkennung.

4.2 Objekterkennung

Den Begriff der Objekterkennung verbindet man insbesondere mit der Objekterkennung innerhalb von digitalen Bildern. Dies beschreibt aber nur ein Teilgebiet der Objekterkennung, da es nicht nur visuelle Verfahren zur Objekterkennung, sondern auch

nicht visuelle Verfahren gibt. Auf diese Unterteilung wird insbesondere nochmal im Abschnitt 4.2.2 eingegangen, da insbesondere in diesem zwischen visuellen- und nicht visuellen Tracking Verfahren unterschieden wird.

Bevor die Tracking Verfahren genauer untersucht werden, wird zunächst das Thema der Detektion behandelt, bei dem die Position eines Objekts zwar nicht eindeutig erkannt wird, aber zumindest seine Anwesenheit festgestellt werden kann.

Abschließend wird zur Komplementierung des Abschnittes noch das Thema der Identifikation behandelt, welches für die Objekterkennung im Rahmen dieser Arbeit benötigt wird. Damit unterscheidet sich die Definition für Objekterkennung auch von anderen Arbeiten, welche Objekterkennung lediglich als die Identifikation des Objekttyps (beispielsweise Mensch, Baum, oder Tier) ansehen. In dieser Arbeit wird unter Objekterkennung, das folgende verstanden:

Objekterkennung beschreibt ein Verfahren, welches Objekte innerhalb eines Objektraumes identifizieren kann. Neben der eindeutigen Identität des Objektes wird auch dessen Position im Objektraum ermittelt.

4.2.1 Detektion

In erster Linie denkt man bei dem Begriff Detektion an einen Detektor, ein Gerät mit welchem die Anwesenheit eines bestimmten Stoffes oder Vorganges nachgewiesen werden kann. Ein Beispiel sind Metalldetektoren. Mit diesen kann zwar die Anwesenheit von Metall festgestellt werden, aber nicht seine genaue Position.

Anhand dessen, lässt sich die Abgrenzung zum Tracking am besten beschreiben, da die Position eines detektieren Objektes sich im Unterschied zu einem Getrackten, lediglich in einem bestimmten Bereich angeben (ähnlich wie beim GPS-Tracking mit lediglich einem Satelliten, siehe 4.2.3.1) lässt.

Da der Begriff im Sprachgebrauch nicht ganz eindeutig ist, folgt zunächst eine Definition und anschließend eine Abgrenzung zum Tracking.

„Detektion (englisch: detection) ist vom lateinischen „detegere“ abgeleitet und heißt so viel wie „Entdeckung“. Vor allem in naturwissenschaftlichen Bereichen verwendet, bedeutet Detektion das Entdecken, Erfassen, Isolieren, Feststellen oder das Sichtbarmachen.“[Dete00].

Ein einfaches Beispiel hierfür zeigt ein Blick auf die Sendeverfolgung von Paketdienstleistern. Ein zu versendendes Paket wird immer nur in bestimmten Abständen detektiert. Zum Beispiel wenn es im Paketshop zum Versenden eingeht, wenn es sich in einem Umsortierungszentrum befindet oder beim Empfänger ankommt. In der Zwischenzeit können zwar grobe Angaben über seinen derzeitigen Aufenthaltsort gemacht werden, nicht aber eine exakte Positionsangabe.

Darüber hinaus lässt sich beim Detektieren nicht zwangsläufig auch die Identität bestimmen, im vorherigen Beispiel, war dies zwar der Fall (da die Identität der Pakete bekannt war), aber im folgenden Beispiel anhand der in Kapitel 3.1 vorgestellten Modelfabrik nicht. Diese besitzt neben den RFID Köpfen, welche die rein visuell nicht unterscheidbaren Werkstücke eindeutig anhand ihrer ID unterscheiden können, noch Lichtschranken, die lediglich kurzzeitig ihre Anwesenheit detektieren können. Dementsprechend kann nur anhand der Informationen der Lichtschranken, weder die Position noch die Identität des Werkstückes eindeutig beschrieben werden. Sie dienen lediglich als zusätzliche Informationen, welche in Kombination mit weiteren Informationen aufbereitet werden können (siehe dazu auch Abschnitt 4.3 Informationsfusion).

4.2.2 Tracking

Die Bedeutung des Trackings wurde im Abschnitt 4.1 bereits kurz erläutert. Im Gegensatz zur Detektion ist Tracking nicht nur für die Objekterkennung von großer Bedeutung sondern auch für Mixed-Reality Anwendungen, da es für das Device von großer Bedeutung ist, seine eigene Position, sowie seine Umwelt, zu kennen, um AR-Objekte passend im Raum zu positionieren (siehe auch Abschnitt 3.5.1 SLAM). Für die physikalischen Objekte hingegen bietet es eine Möglichkeit der Objekterkennung. Aufgrund dessen wird sich im Folgenden zunächst mit den theoretischen Grundlagen von Tracking und anschließend mit einigen Verfahren hierfür beschäftigt.

Der Begriff Tracking im allgemeinen Sprachgebrauch unscharf und besitzt teilweise eine andere Bedeutung als im hier verwendeten Kontext von Mixed Reality Anwendungen und Objekterkennung. Deswegen folgt zunächst eine Definition des Begriffes:

„Unter Tracking (dt. verfolgen) versteht man gemeinhin die Berechnung oder korrekter eigentlich die Schätzung der Position und/oder Lage/Orientierung). Im Fall von AR ist es erforderlich, zu jedem Zeitpunkt den Blickpunkt des Betrachters so genau wie möglich zu erfassen. Da die Realität jedoch zumeist in Form des zuvor aufgenommen Videobildes vorliegt, schätzt man stattdessen zumeist die Position und Lage der verwendeten Kamera.“ [DBGJ14 S. 242f.]

Heißt zusammenfassend, dass es sich bei Tracking Verfahren um Verfahren zur Positionsbestimmung handelt.

Ein weiterer Aspekt der beim Tracking beachtet werden sollte, ist die Genauigkeit. Diese hängt insbesondere vom geplanten Anwendungsbereich der MR-Applikation ab und spielt dementsprechend einen bedeutenden oder weniger bedeutenden Aspekt. Im Folgenden werden einige Anwendungsbereiche von MR im Zusammenhang mit ihrem Anspruch an die Genauigkeit betrachtet [MeRe11]:

- Gaming: Hierbei handelt es sich für die breite Masse um den populärsten Anwendungsbereich. MR Spiele wie beispielsweise Pokemon GO benötigen zwar die Möglichkeit ihre Realität abbilden zu können, die Genauigkeit ist aber lediglich ein optionales Kriterium zur Erhöhung des Spielspaßes, nicht aber sicherheitskritisch.
- Industrielle Anwendung: Abhängig von der Anwendung und der Produktionsumgebung wird eine hohe Genauigkeit erfordert. Werden beispielsweise Anlagen- oder Gebäude Komponenten innerhalb der vorhandenen Umgebung geplant und eingepasst, führt Ungenauigkeit ggf. zu Planungsfehlern.
- Medizin: Hier ist eine sehr hohe Genauigkeit erforderlich, insbesondere bei Anwendungen im Bereich der Chirurgie, bei denen mithilfe von MR-Informationen direkt auf den Patienten abgebildet werden [Augm00c].

Grundsätzlich hängt der Anspruch der Genauigkeit aber immer von der tatsächlichen Anwendung und nicht nur vom Einsatzbereich ab, da auch in der Medizin und der Industrie Anwendungen vorstellbar sind, welche lediglich Informationen an die Wand projizieren, welche nicht im Kontext zu einem realen physikalischen Objekt stehen.

Nachdem bisher die Grundlagen für das Tracking erörtert wurden, werden im Folgenden konkrete Verfahren und Möglichkeiten für das Tracking von bewegten Objekten erörtert.

Zunächst gilt es zwischen den folgenden 2 Verfahren zu unterscheiden, welche dazwischen unterscheiden, ob das Objekt selbst über die Möglichkeit verfügt seine Position zu ermitteln, oder ob die Position ermittelt wird [MeRe11 K3.1 ff]:

- Inside-Out: Hierbei ermitteln die bewegten Objekte¹⁷ selbst ihre Positionen oder enthalten diese.
- Outside-In: Die bewegten Objekte verfügen über keinerlei Informationen über ihre Positionen

Das folgende kurze Beispiel dient dazu den Unterschied der beiden besser nachzuvollziehen:

Mithilfe der HoloLens und eines Bilderkennungsframeworks (z.B. Vuforia) wird die Position eines Werkstückes im Raum ermittelt.

Die HoloLens verfügt hierbei über die Kenntnis seiner eigenen Position, sowie des Abbildes des Raumes (Inside-Out). Das Werkstück hingegen befindet sich zwar in diesem Raum, verfügt, selbst aber über keinerlei Informationen seiner eigenen Position. Die Position wird lediglich von der HoloLens mithilfe einer Bilderkennung ermittelt (Outside-In).

Inside-Out Verfahren werden in der Regel von AR-Devices angewendet, da diese über eine Vielzahl von Sensorik verfügen und die Kenntnis über ihre eigene Position essenziell für Mixed Reality Anwendungen ist.

Darüber hinaus können selbstverständlich auch andere Objekte ihre eigene Position über ein Inside-Out Verfahren ermitteln und an das AR-Device weitergeben. Hierfür wird aber zunächst eine zusätzliche Verbindung zwischen dem AR-Device und dem physikalischen Objekt benötigt und anschließend müssen die unterschiedlichen Koordinatensysteme des Devices und des Objektes miteinander synchronisiert werden, da es ansonsten zu einer falschen Annahme der Position innerhalb des AR-Devices kommen könnte. Überdies ist der technische Aufwand für die meisten Objekte um diese mit eigener Sensorik auszustatten, zu hoch und nicht zielführend, vor allem in industriellen Anlagen, in

¹⁷ Beachte, dass auch das MR-Device an sich ein bewegtes Objekt ist.

welchen die Werkstücke extremen Bedingungen innerhalb der Fertigung ausgesetzt wären (z.B. Hitze, oder wie im Beispiel einem Lackierungsprozess).

Deswegen werden für das Tracking von Objekten in der Regel Outside-In Verfahren verwendet, bei denen das Objekt an sich keinerlei Kenntnisse über seine eigene Position hat.

Neben diesen beiden Verfahren wird beim Tracking noch zwischen visuellem und nicht visuellem Tracking unterschieden. Dies wird in den folgenden beiden Abschnitten genauer erläutert.

4.2.3 Nichtvisuelles-Tracking

Unter einem visuellen Verfahren wird eine Tracking Methode unter Zuhilfenahme eines Sensors zur Bildverarbeitung bezeichnet, während nichtvisuelle Sensoren gänzlich auf solche Methoden verzichten.

4.2.3.1 GPS

Das bekannteste nichtvisuelle Verfahren zur Positionsermittlung ist hierbei vermutlich das Global Positioning System (kurz GPS). Hierbei handelt es sich um ein Ortungssystem, welches auf Satelliten Technik basiert. Im alltäglichen Sprachgebrauch wird GPS aber vor allem mit Navigationssystemen verbunden, da diese den größten Anwendungsbereich hierfür darstellen.

Die zugrunde liegende Technologie hierbei misst die Reichweite zwischen einem GPS-Empfängergerät und den dazugehörigen Satelliten zu messen. Aus diesen Reichweiten und den bekannten Positionen der Satelliten kann schlussendlich die genaue Position des Empfängers ermittelt [XuXu16 S.1].

Abbildung 17 zeigt hierbei das Grundprinzip hierfür. Prinzipiell sind für eine Positionsbestimmung mit GPS 3 Satelliten notwendig, welche ständig eine Nachricht mit ihrer aktuellen Id, Zeit sowie dem Zeitpunkt zu welchem die Nachricht versendet wurde, versenden. Der GPS-Empfänger vergleicht die Nachrichten und berechnet mittels Trilateration ¹⁸ die genaue Position des Empfängers.

¹⁸ Trilateration ist ein Messverfahren zur Positionsbestimmung eines Punktes. Anhand der bekannten Entfernung zu einem Punkt, kann ein Kreis um diesen gebildet werden. Der Schnittpunkt der drei gebildeten Kreise bestimmt dabei die Position des gesuchten Punktes. [Torg91 S. 214 ff.] Siehe auch Abbildung 17.

Moderne GPS Systeme benötigen eine Verbindung zu mindestens 4 Satelliten und bieten neben der Möglichkeit der Positionsbestimmung auch noch Methoden zur Geschwindigkeitsermittlung des Empfängers und dessen Bewegungsrichtung an.

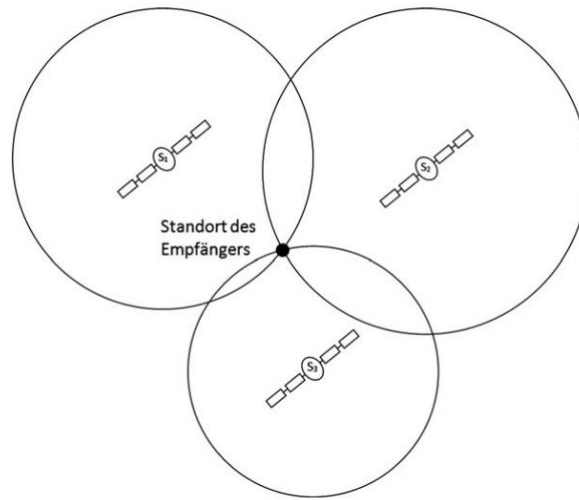


Abbildung 17: Ortung durch Entfernungsmessung in zwei Dimensionen. Der Schnittpunkt der 3 Kreise der jeweiligen Abstände zum Satellit S1-S3 beschreiben die aktuelle Position des Empfängers. [Schü14]

4.2.3.2 Weitere nicht visuelle Trackingverfahren

GPS bietet eine Ortungslösung welche nahezu weltweit eingesetzt wird, insofern der Empfänger nicht von den Satelliten abgeschirmt ist. Für Mixed-Reality Anwendungen ist dies aber einerseits zu überdimensioniert, da die Objekte sich i.d.R. lediglich in einem bestimmten Bereich bewegen und nicht weltweit, und andererseits zu ungenau sein könnte (abhängig vom GPS System [GPS, A-GPS etc.] beträgt die Genauigkeit zwischen 1-15m).

Für MR-Anwendungen ist es daher eher entscheidend die Positionsbestimmung innerhalb eines bekannten Gebietes vorzunehmen. Mithilfe eines bekannten Abbildes des Raumes und einem oder mehreren Sensoren zur Abstandsmessung, kann die aktuelle Position innerhalb dieses ermittelt werden. Typische Sensoren zur Abstandsmessung sind unter anderem [Zhan03]:

- Infrarotsensoren
- Ultraschallsensoren
- Lasersensoren

Die Abstandsermittlung hierbei erfolgt entweder über die Signallaufzeit (engl. Time-of-flight), die Phasendifferenz, oder Triangulation.

Für das Messen der Signallaufzeit wird zunächst ein Impuls ausgesendet und anschließend die Zeit bis zum Eingang eines Echos gemessen. Der Abstand ergibt sich daraufhin wie folgt [Zhan03]:

$$Abstand = \frac{\text{gemessene Zeit} * \text{Geschwindigkeit im Medium}}{2}$$

Bei der Messung der Phasendifferenz hingegen wird ein Laserstrahl mit einer bekannten Periodendauer T zunächst in zwei Strahle aufgesplittet. Der Referenzstrahl wird von einem Phasenmessgerät in der Entfernung L gemessen und der zweite Strahl läuft bis zur Objektoberfläche und zurück zum Phasenmessgerät. Nachdem die Phasendifferenz (φ) gemessen wurde, kann mithilfe der folgenden Beziehung die Zeitdifferenz ermittelt werden [Oubb12] :

$$\frac{\Delta t}{T} = \frac{\varphi}{2\pi}$$

Bei der Triangulation wird mithilfe der trigonometrischen Beziehung und zwei Sichtpunkten, welche in einem bekannten Abstand zueinander stehen der jeweilige Winkel zum Objekt bestimmt und daraus anschließend die Distanz bestimmt.

Im Folgenden werden noch einige weitere nicht visuelle Sensoren aufgelistet, welche nicht unbedingt zur Abstandsermittlung dienen, aber sonstige wichtige Informationen ermitteln.

- Kompass: Über das Magnetfeld der Erde wird die Ausrichtung des Devices bestimmt [MeRe11 K. 31.1.2]
- Gyrosensor: Ein Beschleunigungs-oder Lagesensor, welche auf kleinste Beschleunigungen, Drehbewegungen oder Lageänderungen reagiert. [Gyro00]

Durch die Kombination der ständig aufgezeichneten Informationen, kann das Device/Objekt seine Position bestimmen. Genauer zur Kombination unterschiedlicher Informationen wird im Abschnitt Informationsfusion behandelt.

4.2.4 Visuelles-Tracking

Diese Tracking Verfahren basieren im Gegensatz zu den nicht visuellen auf einem visuellen Sensor, i.d.R. einer Videokamera [MeRe11 K. 3.1.1.3].

Hierbei wird zunächst zwischen dem Verhalten der Kamera unterschieden, also ob es sich um eine fest verbaute Kamera (z.B. eine feste Webcam) oder eine frei bewegliche (z.B. ein HMD oder ein Smartphone) handelt. Insofern es sich um eine fest verbaute Kamera

handelt muss der Tracker lediglich mithilfe einer Bildverarbeitungsroutine die Position der realen Objekte ermitteln, während hingegen bei freien Devices weitere Informationen benötigt werden [MeRe11 K. 3.1.1.3]. Hierbei könnte es sich zum Beispiel um Informationen der nicht visuellen Sensoren des vorangegangenen Abschnittes handeln.

Darüber hinaus wird zwischen Marker- und Merkmalsbasiertem visuellen Tracking unterschieden. Beide Verfahren werden im Folgenden genauer vorgestellt.

4.2.4.1 Markerbasiertes Tracking

Markerbasierende Verfahren wurden bereits Ende der 90er Jahre für MR Anwendungen dargestellt und bilden derzeit noch immer ein weitverbreitetes Verfahren.

Bei einem Marker handelt es sich häufig um ein schwarz-weißes Muster, welches sich in einem Videostream leicht identifizieren lässt [DBGJ14 S. 257]. Die folgende Abbildung zeigt einige typische Marker, welche in bekannten AR Frameworks wie dem ARToolkit oder ARTag eingesetzt werden.



Abbildung 18: Beispiele für Marker [DBGJ14 Abb. 8.13]

Der Marker selbst ist dem Tracking-Device bekannt und dieses sucht innerhalb seines Bildbereiches nach zusammenhängenden Liniensegmenten. Diese werden anschließend mit dem vordefinierten Marker verglichen und wenn ein Marker gefunden wird, berechnet das Device seine Position und Lage im Abstand zum gefundenen Marker.

Anhand der folgenden Abbildung sollen zunächst die Zusammenhänge zwischen dem Marker und der Kamera deutlich gemacht werden und anschließend die Theorie für die Berechnung der Position hergeleitet werden

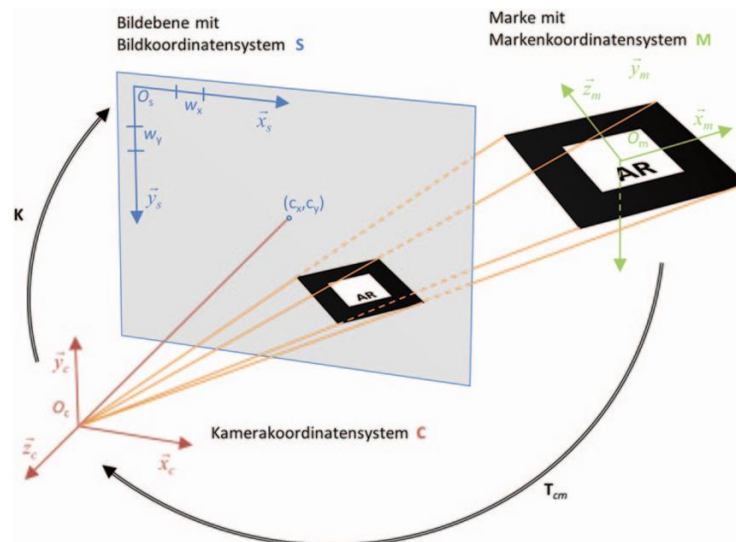


Abbildung 19: Zusammenhang zwischen Kamerakoordinatensystem C und dem Markerkoordinatensystem M . [DBGJ14 S.260]

Das Verhältnis zwischen dem Marker und der Kamera basiert auf Basis Marker Eckpunkte und der Tatsache, dass die Größe des Markers bekannt sein muss. Der Zusammenhang zwischen dem Marker- und dem Kamerakoordinatensystem (M und C) beschreibt die Matrix T_{cm} , welche sich aus einer 3×3 Rotationsmatrix R und einem Translationsvektor \vec{t} mit jeweils drei Freiheitsgraden zusammensetzt. Mithilfe der Transformationsmatrix berechnet sich eine Koordinate v_m des Markerkoordinatensystems M zu v_c im Kamerakoordinatensystem C :

$$v_c = T_{cm} \cdot v_m$$

Für weitere Informationen siehe: [DBGJ14 S. 259 f.]

Die genauen Trackingverfahren unterscheiden sich hierbei zwar in den unterschiedlichen Frameworks, genauso wie die Vorgaben an den Marker (Form, Farbe, Abmessung), basieren aber auf den selben mathematischen Grundlagen. Je nach Framework ist es zum Beispiel wichtig wie viel Prozent des Markers für eine Identifikation erkannt werden müssen.

Die Vorteile des Einsatzes von Markern sind vor allem, dass diese leicht per Ausdruck erstellt werden können und auf den unterschiedlichsten Objekten, Wänden, in Büchern oder anderen Medien untergebracht werden können und damit leicht zu verbreiten. Da Marker im Gegensatz zu QR- oder Barcodes keine Informationen kodieren, ist der Anwender darüber hinaus frei in der Verwendung.

Als Nachteile lassen sich dahingegen anbringen, dass die Marker einerseits zunächst an den Objekten angebracht werden müssen (Aufwand) und an der Ausrichtung des AR-Objektes, da dieses oft nur in einer Blickposition optimal dargestellt werden kann. Darüber hinaus sind sie nicht für jede Produktionsumgebung geeignet.

[vgl. DBGJ14 S.256 ff.]

4.2.4.2 Merkmalbasiertes Tracking

Merkmalbasierte Verfahren basieren vom Prinzip auf der gleichen Idee wie die Markerbasierten. Anstelle eines Markers wird jedoch nach anderen bekannten Merkmalen innerhalb des Bildbereiches gesucht. Hierbei kann es sich zum Beispiel um ein bekanntes 2D- oder 3D-Modell handeln. Ein bekanntes AR Framework welches dies unterstützt wird in Kapitel 6.2.4 genauer vorgestellt. Hierbei wird das Bild anstelle mit eines Markers mit einem hinterlegten 3D/2D-Modell verglichen. Dabei werden aus dem Kamerabild heraus die Ecken und Kanten extrahiert und anschließend die Abstände zwischen diesen errechnet, siehe Abbildung 20 und für weitere Informationen: [ReDr06]



Abbildung 20: Übersicht über den Tracking Prozess mithilfe eines 3D Modelles

Neben dem geometriebasierten Tracking welches insbesondere auf die Ecken und Kanten eines bekannten 3D Modelles zurückgreift, gibt es noch Verfahren welche auf andere visuelle Merkmale in Bildern zurückgreifen. Diese sind häufig für das menschliche Auge nicht ohne weiteres erkennbar, aber für Merkmalsdetektoren, welche mit verschiedenen Filtertechnologien der Bildbearbeitung funktionieren, leicht zu identifizieren. Dabei kommt es insbesondere auf den Detailreichtum des Bildes, je detailreicher das Bild ist, desto leichter ist es für das Trackingframework Merkmale dieses wiederzuerkennen. Weitere Informationen hierzu finden sich in Abschnitt 6.2.4, welcher das Vuforia Framework vorstellt, siehe insbesondere auch: Kapitel 7.1.2 Visuell mithilfe von Vuforia.

Eine weitere Form bildet die Erkennung von Gesichtern, auch als Face Tracking bezeichnet. Dieses wird bereits seit ca. 30 Jahren erforscht und wird nicht nur im MR Bereich verwendet, sondern unter anderem auch noch im Entertainment Bereich, z.B. für

Video-Spiele) oder zur Identifikation bei Pässen oder aber auch an zur Anmeldung an Rechnern oder anderen Sicherheitskontrollen. [MeRe11 K. 3.1.4].

Die Vor- und Nachteile der markerbasierten Anwendungen hängen letzten Endes natürlich immer von der konkreten Ausprägung ab, aber ein entscheidender Vorteil gegenüber den Markern ist, dass Verfahren wie beispielsweise Objekterkennung anhand von 3D Modellen keine Modifikation des zu identifizierenden Objektes mehr benötigen, das heißt es muss kein Marker oder ähnliches darauf geklebt werden. Damit ist das Objekt nicht mehr in seinen Einsatzumgebungen eingeschränkt.

Des Weiteren sind einige merkmalsbasierte Verfahren, z.B. ein SURF¹⁹ basierter Ansatz um einiges Robuster als Markerbasierte [DBGJ14 S. 263]. Merkmalsbasierte Verfahren werden im Normalfall für die Lösung des SLAM-Problems mit visuellen Sensoren eingesetzt.

Der große Nachteil dieser Verfahren ist der erhöhte Rechenaufwand. Im Vergleich zu einer einfachen Marker-Erkennung müssen viel mehr Berechnungen im Hintergrund erledigt werden, die von einigen Endgeräten (z.B. ältere Smartphones) nicht erbracht werden können.

4.2.5 Identifikation

Die vorherigen Verfahren weisen bisher immer noch ein Problem auf und zwar, dass sie zwar den Typ oder die Position eines Objektes bestimmen können, aber in der Regel nicht die eindeutige Identität dieses Objektes. Insbesondere die visuellen Verfahren, können innerhalb eines Bildes beispielsweise ein Haus oder einen Menschen erkennen, aber ihm bisher noch nicht seine Identität zuordnen. Genauso wenig wie die Lichtschranken der Modelfabrik, welche in Abschnitt 4.2.1 erwähnt wurden.

In diesem Beispiel wurde auch bereits eine Möglichkeit zur Identifikation von Objekten mitgeliefert, nämlich mithilfe der RFID Reader. Der folgende Abschnitt soll sich nun mit weiteren Möglichkeiten der Objektidentifikation beschäftigen.

RFIDs-Chips bieten bereits eine High-Tech Lösung für dieses Problem an, allerdings gibt es hier noch einige einfachere Verfahren.

¹⁹ Speeded Up Robust Features, ist ein Algorithmus zur schnellen und robusten Erkennung von Bildmerkmalen für maschinelles Sehen. [Surf00]

Die vermutlich einfachste Möglichkeit, wäre es die Information (bzw. die ID) einfach an das physikalische Objekt heranzuschreiben.

Diese Möglichkeit lässt sich ebenfalls leicht auf Mixed-Reality Anwendungen übertragen, indem die Position des physikalischen Objektes in der Anwendung einprogrammiert²⁰ wird. Die zu dem physikalischen Objekt gehörenden Informationen (oder zumindest die genaue ID) werden dann innerhalb der MR immer passend an derselben Stelle wie das physikalische Objekt angezeigt. Dieses recht simple Verfahren bietet eine zuverlässige Lösungsmöglichkeit, solange die Objekte ihre Position nicht verändern. Aufgrund dessen eignet es sich nicht für dynamische Elemente, wie die Werkstücke innerhalb der Modelfabrik.

Andere mögliche Identifikationsverfahren, basieren wie der RFID-Chip darauf, dass die Information über die Identität in irgendeiner Form an dem Objekt hinterlegt wird und über eine geeignete Schnittstelle extrahiert wird. Für AR-Reality Devices bieten sich hier insbesondere optische Identifikationsverfahren an, wie beispielsweise Bar- oder QR-Codes. Bei den Bar-Codes enthalten ihre Informationen eindimensional, wohingegen die Informationen des QR-Codes in einer zweidimensionalen Matrix gespeichert werden. Beide sind in der folgenden Abbildung 21 zu sehen.



Abbildung 21: Abbildung eines Bar-Codes (links) sowie eines QR-Codes (rechts).

Ein möglicher Nachteil der Bar- und QR-Codes ist, dass diese immer großflächig an dem Produkt angebracht werden müssen. Dies könnte aus Gründen der Ästhetik, des Aufwandes oder sonstigen Motiven unerwünscht sein. Insbesondere wenn in industriellen Anlagen größere Stückzahlen eindeutig unterschieden werden sollen, könnte dies zu einem erheblichen Mehraufwand führen.

4.3 Informationsfusion

In den bisherigen Abschnitten wurden Verfahren dargestellt um die Position, die Anwesenheit oder Identität eines Objektes zu ermitteln. Jedes Verfahren für sich alleine konnte dabei oftmals, oder zumindest nur eingeschränkt, das Problem der Objekterkennung lösen. Die Kombination dieser Verfahren bietet hierfür jedoch eine

²⁰ Innerhalb der Programmiersprache Unity wird dies als Anchor (engl. für Anker bezeichnet).

Möglichkeit und wurden in den vorangegangenen Abschnitten bereits angeschnitten aber nicht ausführlich definiert.

Zusammengefasst wird dies unter dem Begriff Informationsfusion, welcher Methoden umfasst, die Daten aus unterschiedlichsten Daten- und Informationsquellen kombinieren, um präzisere Angaben über die Ergebnisse von Messwerten machen zu können. In der Literatur wird neben dem Begriff Informationsfusion auch Daten- oder Sensorfusion verwendet. Teilweise werden die Begriffe simultan verwendet, aber teilweise wird unter Sensorfusion auch nur die reine Fusion von Sensordaten und unter Datenfusion die reine Datenfusion behandelt [DGJS02].

Da für MR-Anwendungen Daten aus unterschiedlichsten Datenquellen und nicht nur aus Datenbanken oder nur aus Sensoren von Interesse sind, wird im Rahmen dieser Arbeit nur das Thema Informationsfusion behandelt und wie folgt definiert:

Mit Informationsfusion wird der Prozess bezeichnet, Daten aus unterschiedlichen Sensoren oder Informationsquellen mit dem Ziel zu verknüpfen, neues oder präziseres Wissen über physikalische Größen, Ereignisse und Situationen zu gewinnen [RuPu06].

Mit dieser Definition wird der Schwerpunkt des Fusionsergebnisses insbesondere auf die Qualität der Fusionsergebnisse gelegt. Informationsfusion ist bereits heute ein integraler Bestandteil vieler technischer und biologischer Systeme und hat vielseitige Einsatzgebiete, u.a. in der Messtechnik, Militärtechnik, Robotik Navigation, Fahrerassistenzsystemen etc. [RuPu06].

Letztgenannte bilden hierbei ein wunderbares Beispiel für Informationsfusion. Ein Fahrerassistenzsystem bezeichnet die technische Unterstützung des Fahrers im Fahrprozess, d.h. bei der Stabilisierung, Führung und Navigation des Fahrzeuges [NGWS05]. Ein Fahrerassistenzsystem besteht hierbei aus mehreren Fahrerassistenzfunktionen. Hierbei handelt es sich unter anderem um ein ACC²¹-System, welches den konstanten Abstand zum vorausfahrenden Fahrzeug hält. Eine weitere Komponente eines Fahrerassistenzsystems könnte z.B. der automatische Spurhalteassistent LDW²². Beide Systeme basieren auf unterschiedlichen Sensoren, z.B.

²¹ ACC: Adaptive Cruise Control.

²² LDW: lane departure warning system engl. Für Spurassistent.

visueller Sensorik zur Erkennung der Spur und Ultraschall-Sensoren um den Abstand zum vorausfahrenden Fahrzeug zu messen. Um ein sicheres Fahrerassistenzsystem gewährleisten zu können müssen die unterschiedlichen Daten miteinander mittels Informationsfusion kombiniert werden[NGWS05].

Das Konzept der Informationsfusion ist ebenfalls von großer Bedeutung für MR-Anwendungen. Für die Orientierung im Raum beispielsweise kombiniert die HoloLens die folgenden Sensoren [Howh00] :

- Gyrosensor
- Magnetometer²³
- Beschleunigungssensor
- Lichtintensitätssensor
- Tiefensensor
- Graustufen und RGB Kameras

Hierbei werden bisher vor allem Sensordaten miteinander kombiniert, u.a. als Lösung für das SLAM Problem. Für industrielle MR-Anwendungen bietet es sich daher an die Grundidee der Informationsfusion weiter zu verwenden um damit Objekte zu tracken.

4.4 Zusammenfassung

In diesem Kapitel wurde zunächst der aktuelle Stand der Forschung im Bereich Augmented Reality vorgestellt und abschließend eine Vielzahl von Verfahren zur Detektion, Ortung und Identifikation von Objekten vorgestellt. Da die meisten Verfahren für sich alleine gestellt, das Problem der Objekterkennung nicht lösen können, wurde abschließend noch auf das Thema Informationsfusion eingegangen. Die Kombination mehrerer vorgestellter Verfahren, kann es ermöglichen, ein Objekt nicht nur eindeutig zu identifizieren, sondern auch seine Position im Raum zu ermitteln.

Dabei wurde insbesondere auch darauf geachtet, die Verfahren hinsichtlich ihrer Eignung für MR-Anwendungen zu begutachten. Hierdurch wurde GPS beispielsweise als Tracking Möglichkeit für industrielle MR-Anwendungen ausgeschlossen, da diese meistens Indoor- und nicht Outdoor Navigation erfordern. Darüber hinaus ist die benötigte Genauigkeit von wenigen cm, hierbei nicht gegeben. Dafür MR-Anwendungen

²³Ein Sensor zur Messung der magnetischen Flussdichte, in diesem Fall Vergleichbar mit einem Kompass.

Erkennungsgenauigkeiten des Trackings im cm Bereich essenziell sind, wurde in diesem Kapitel auch nicht auf Tracking Verfahren via Bluetooth oder WiFi eingegangen.

Da primär im Rahmen dieser Arbeit die Zuordnung von Informationen zu ihren dazugehörigen Objekten im industriellen Kontext betrachtet wird, werden in der folgenden Lösung insbesondere Verfahren berücksichtigt, welche keine Modifikation an den Objekten erfordern. Die zu trackenden Objekte werden dementsprechend selbst keine Kenntnis über ihre Position haben (Outside-In-Verfahren).

Da die HoloLens bereits als Hardware für den Prototyp ausgewählt wurde muss das SLAM-Problem für das AR-Device nicht weiter behandelt werden. Die HoloLens verfügt zwar neben der Kamera noch über weitere Sensorik, wie beispielsweise den Tiefenscanner, aber diese dienen insbesondere dazu, dass sich das Device im Raum orientieren kann. Folglich wird auf ein visuelles Verfahren für das Objekttracking zurückgegriffen. Abschließend werden zur eindeutigen Identifikation des Objektes Ansätze aus dem Abschnitt Informationsfusion verwendet.

5 Lösungskonzept

Ausgehend von der in Kapitel 3 beschriebenen Problemstellung und auf Basis des aktuellen Standes der Forschung wird in diesem Kapitel ein Lösungsvorschlag für das herausgearbeitete Kernproblem der Objekterkennung erarbeitet. Die anderen in Kapitel 3 herausgearbeiteten Probleme wie SLAM und das Zuordnen der Datenquellen werden in diesem Kapitel nicht weiter beschrieben, da sie nicht Kern der Arbeit sind, aber in Kapitel 6 Implementierung kurz anhand eines Prototyps erläutert werden.

In Abbildung 22 werden die herausgearbeiteten Probleme aus Kapitel 3 nochmals zusammengefasst und im Kontext eingeordnet. Die grauen Probleme werden wie bereits erwähnt nicht weiter behandelt und die beiden Kernprobleme (Objekterkennung und die Positionierung des MR Objektes) sollen allgemein mit einem allgemeinen Lösungspattern beschrieben werden können. Dies wird in der Abbildung mit dem Position Mapper bereits angedeutet und im Folgenden weiter beschrieben.

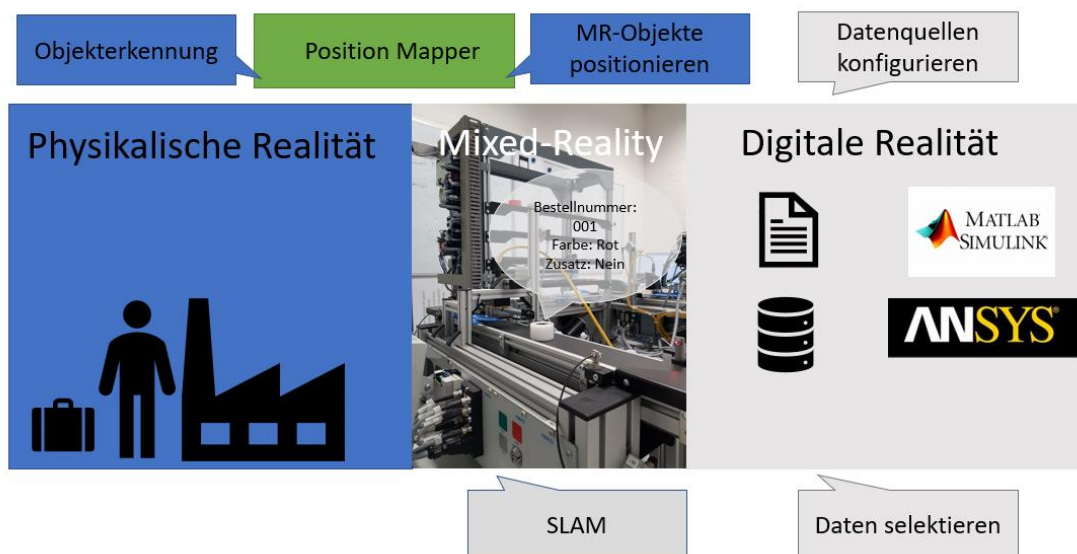


Abbildung 22: Position Mapper als Lösungsmethode für die Probleme der physikalischen Welt und der Mixed Reality

5.1 Position Mapper Pattern

Ziel dieses Pattern ist es, das Problem der Objekterkennung zu lösen. Das heißt, dass ein zu getracktes Objekt nicht nur anhand seines Typs z.B. Werkstück oder Mensch erkannt wird, sondern dass dazu auch seine Identität bestimmt wird (z.B. Werkstück Nr. 5, Mensch: Max Mustermann). Für die Lösung dieses Problems wurde das in Abbildung 23 dargestellte Pattern entworfen.

Dies basiert auf den Annahmen, welche in Kapitel 3 Problemstellung für physikalische Objekte herausgearbeitet wurden. Diese haben entweder eine statische (feste) oder eine dynamische (variable) Position. Des Weiteren können die physikalischen Objekte in Typen unterteilt werden.

5.1.1 Lösung

Für die Lösung des Problems erkennt der Position Mapper zunächst mithilfe eines geeigneten Trackingverfahrens (siehe auch Tracking) die Position eines zu identifizierenden Objektes und detektiert seinen Typ (siehe auch Detektion).

Das physikalische Objekt selbst muss sich innerhalb einer definierten Strecke befinden. Zu Beginn einer Strecke befindet sich jeweils ein Synchronisierungspunkt. Hierbei handelt es sich um eine feste Stelle an der das physikalische Objekt eindeutig identifiziert wird. Die Identität des Objekts wird dann anschließend auf der Strecke gespeichert, bis es den Synchronisierungspunkt am Ende der Strecke passiert und diese wieder verlässt.

Der Position Mapper kann anschließend basierend auf diesen Informationen und dem bekannten Verhalten eines Objektes seine Identität bestimmen.

5.1.2 Struktur

Die Struktur des Patterns wird in Abbildung 23 dargestellt und in zwei Welten unterteilt. Auf linken Seite des Trennstriches befindet sich die physikalische Welt, welche das **physikalische Objekt** in den Fokus stellt und auf der rechten Seite die Mixed Reality. Hier befindet sich das **MR Objekt** an einem Ort innerhalb dieser, dem **MR Ort**.

Der **Position Mapper** bildet hierbei das Bindeglied zwischen beiden Welten. Er trackt die **Position** eines physikalischen Objektes und detektiert gleichzeitig den **Objekt Typ**. Anschließend berechnet er mithilfe des **Verhaltens** und den Informationen über die **Strecke** den **MR Ort**.

Ein physikalisches Objekt kann sich immer an genau einem **Ort** befinden. Hierbei handelt es sich entweder an einem **Synchronisierungspunkt** um eine **Strecke**. **Synchronisierungspunkte** definieren jeweils den Anfang und das Ende einer Strecke und bieten die Möglichkeit, das **physikalische Objekt** zu identifizieren. **Synchronisierungspunkte** und **Strecken** lassen sich dabei mithilfe einer **Position** (einem XYZ-Vektor) beschreiben. Ein Synchronisierungspunkt besteht aus genau einer **Position** während sich eine Strecke aus einer Vielzahl von **Positionen** zusammensetzt.

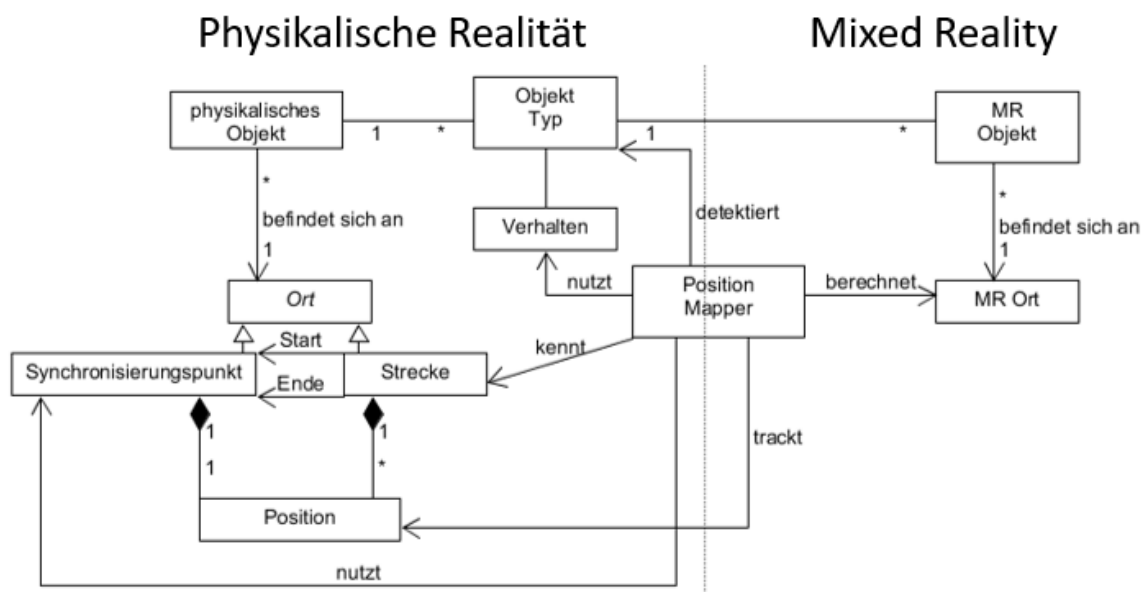


Abbildung 23: Struktur des Position Mapper Patterns

Physikalische Objekte lassen sich darüber hinaus noch **Objekt Typen** zuordnen, welche gleichzeitig aus das **Verhalten** dieses Objektes abbilden. Ersteres beschreibt hierbei eine gemeinsame Oberklasse für die physikalischen Objekte zum Beispiel: **Zylinderförmiges Werkstück**, oder **quadratisches Werkstück** und letzteres charakterisiert die typischen Bewegungseigenschaften. Im einfachsten Fall handelt es sich hierbei um die konstante Fortbewegungsgeschwindigkeit.

5.1.3 Dynamik

Die Dynamik des Systems lässt sich am besten beschreiben indem man zwischen dem Verhalten des Werkstücks und dem des Position Mappers unterscheidet.

Abbildung 24 beschreibt das Verhalten eines dynamischen **physikalischen Objektes**. Die Identität dieses Objektes ist nicht bekannt, solange es noch keinen **Synchronisierungspunkt** passiert hat. Nachdem es am Synchronisierungspunkt identifiziert wurde, bewegt es sich so lange auf der **Strecke** bis es den folgenden

Synchronisierungspunkt am Ende der Strecke passiert hat. In der Zeit, in welcher es sich auf der **Strecke** befindet ist seine Identität bekannt, aber nicht seine genaue Position.

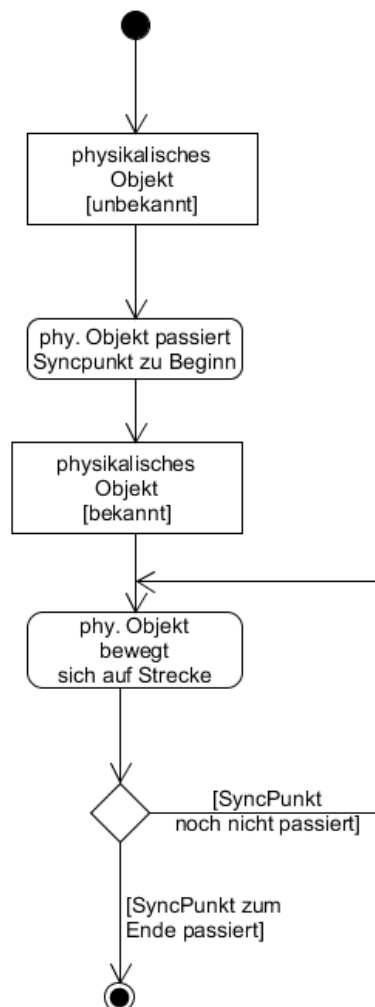


Abbildung 24: Zustandsdiagramm eines dynamischen physikalischen Objektes

Abbildung 25 beschreibt nun den Vorgang wie ein **physikalisches Objekt**, welches sich innerhalb einer bekannten **Strecke** befindet identifiziert werden kann. Hierfür detektiert der **Position Mapper** zunächst ein den **Objekt Typ** eines **physikalischen Objektes** und trackt gleichzeitig seine **Position** dazu.

Im ersten Schritt überprüft der **Position Mapper** ob er einen für ihn bekannten **Objekt Typ** detektiert hat, falls nein ist eine Identifikation nicht möglich. Anschließend wird überprüft, ob sich das getrackte **physikalische Objekt** auf einer bekannten **Strecke** befindet. Falls dies nicht der Fall sein sollte, ist eine Identifikation ebenfalls nicht möglich.

Insofern sich das Objekt auf einer bekannten **Strecke** befindet wird abgefragt wie viele Objekte sich insgesamt auf dieser **Strecke** befinden. Falls lediglich ein **physikalisches Objekt** auf der **Strecke** vorhanden ist, handelt es sich bei diesem um das detektierte **physikalische Objekt** und es kann eindeutig identifiziert werden. Andernfalls wird mithilfe des bekannten **Verhaltens** die genaue Identität und Position des **physikalischen Objektes** bestimmt.

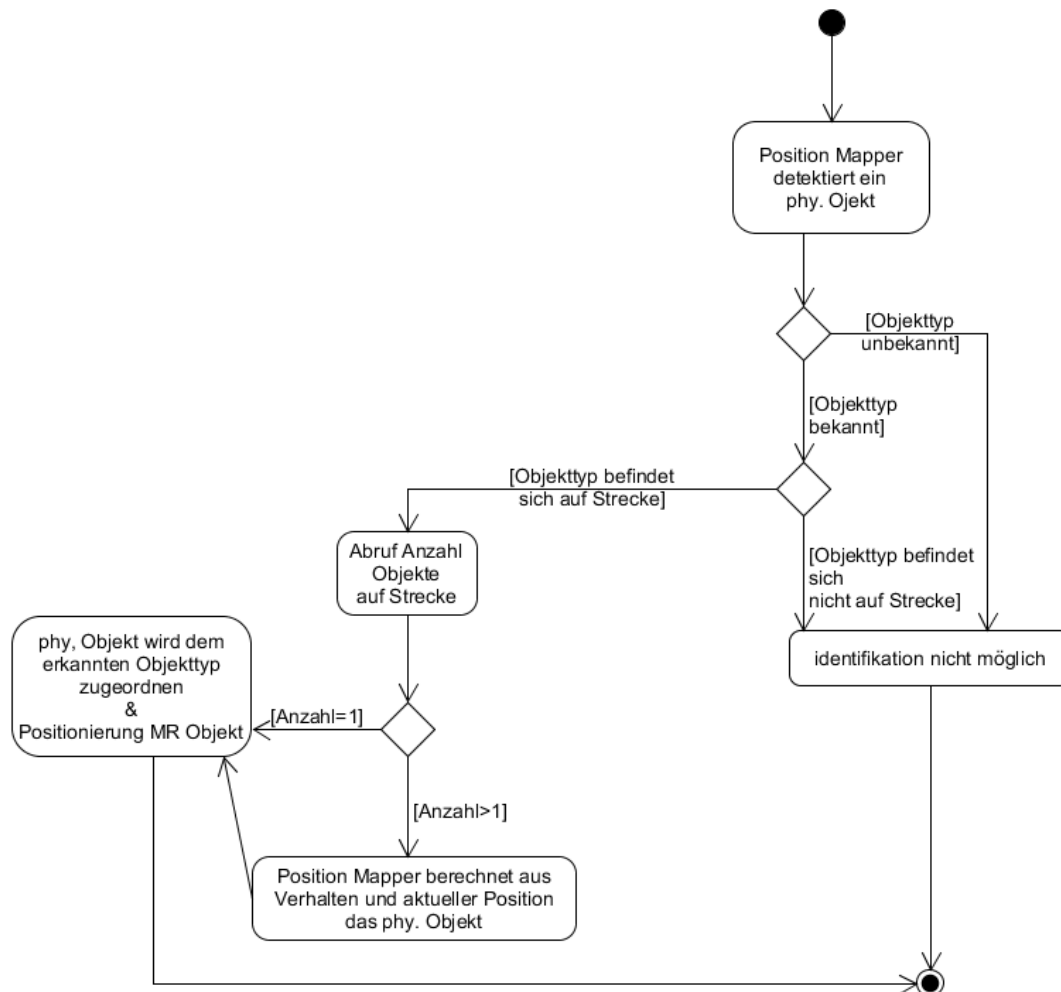


Abbildung 25: Zustandsdiagramm eines Objekterkennungsvorgangs

5.1.4 Zusammenfassung

Dieses Pattern eignet sich insbesondere dazu physikalische Objekte zu identifizieren, welche sich mithilfe eines konventionellen Verfahrens nicht bereits eindeutig identifizieren lassen. Das heißt, dass mithilfe eines Tracking Verfahrens zunächst nur der Objekt Typ bestimmt werden kann. Es wird zum Beispiel nicht ein QR-Code getrackt, welcher bereits die Identität des Objektes enthält.

Darüber hinaus basiert das Pattern auf zwei Annahmen. Zum einen muss sich das Verhalten eines physikalischen Objektes, also sein zu erwartendes Bewegungsprofil beschreiben lassen und zum anderen müssen die Strecken modellierbar sein. Dies bedeutet zum einen das zu Beginn und zum Ende einer Strecke immer ein Synchronisierungspunkt vorhanden sein muss und sich der Weg dazwischen beschreiben lässt, mithilfe von Längenangaben dem Verlauf etc.).

Insofern diese beiden Bedingungen erfüllt sind, eignet sich dieses Pattern zur Objekterkennung bietet eine gute Struktur hierfür an.

6 Implementierung

Im vorangegangenen Kapitel 5 Lösungskonzept wurde eine Möglichkeit vorgestellt, wie Objekte statisch und dynamisch detektiert und identifiziert werden können. Als Proof of Concept wurde dieses Konzept prototypisch im Rahmen dieser Arbeit implementiert. Als MR-Hardware wurde die Microsoft HoloLens ausgewählt und die zu erkennenden Objekte werden durch die Werkstücke der bereits eingeführten Modelfabrik (siehe Kapitel 3.1) repräsentiert. Diese können visuell nicht unterschieden werden, aber anhand ihrer Identität, welche auf dem RFID Chip hinterlegt ist, an bestimmten Stellen (Synchronisierungspunkten) eindeutig identifiziert werden.

Die HoloLens wurde als Hardware gewählt, da diese im Gegensatz zu Smartphones oder Tablets dem Augmented Operator freie Hände ermöglicht und da diese auf ihrer Vielzahl von Sensorik über umfangreiche Möglichkeiten der Umwelterfassung verfügt. Darüber hinaus bringt diese bereits eine Lösung für das SLAM Problem mit sich.

Zu Beginn dieses Kapitels wird zunächst das Gesamtszenario bestehend aus der Modelfabrik, der HoloLens und einem weiteren Rechner als Linker vorgestellt, anschließend werden die zu Grunde liegenden Technologien kurz eingeführt und abschließend die konkrete Implementierung des Prototyps vorgestellt.

6.1 Szenario des Prototyps

Zu Beginn wird in diesem Abschnitt zunächst das Gesamtszenario des Prototyps vorgestellt, welcher zur Evaluation des Position Mappers verwendet wird. Die Abbildung 26 zeigt dafür zunächst einen Gesamtüberblick über das System, bestehend aus der Modelfabrik (links), der HoloLens (rechts), welche die Modelfabrik sieht und beobachtet (angedeutet durch die Grüne Linie), sowie einem Server (Mitte oben). Letzterer dient insbesondere dazu, die Netzwerk Kommunikation zwischen den einzelnen Teilnehmern zu managen.

Auf der Modelfabrik läuft ein Anwenderprogramm, welches wie bereits in Kapitel 3.1 beschrieben, Werkstücke nach individuellen Kundenwünschen „verarbeitet“. Für dieses Szenario wird davon ausgegangen, dass die Werkstücke bereits ihre Informationen (Zielfarbe, Ausprägung, sonstige Kundenwünsche) enthalten und der Augmented Operator, diese Informationen mithilfe der HoloLens an einem Werkstück visualisiert bekommt.

Die Modelfabrik, welche die Werkstücke verarbeitet, besteht aus zwei Siemens S7-300 SPSen, welche die Hardwarekomponenten Hochregal, Förderbänder und die Sensorik steuern. Für eine genaue Darstellung der Modelfabrik sei auf Abbildung 13 und Abbildung 29 verwiesen.

Die HoloLens selbst kommuniziert mit dem Server über den Custom Message Service, welcher Part des MixedRealityToolkits (kurz MRT) ist (siehe auch Abschnitt 6.2.3 f.). Eine direkte Kommunikation zur Modelfabrik findet von der HoloLens aus nicht statt, stattdessen hostet der Server ein weiteres Programm welches mit der Modelfabrik über die Modelfabrik API (siehe auch Abschnitt 6.2.5) kommuniziert. Dies ist notwendig, da die Modelfabrik API in Java geschrieben ist, auf der HoloLens aber keine Java Anwendungen möglich sind. Hierauf wird später genauer im Abschnitt 6.3 Implementierung des Prototyps eingegangen.

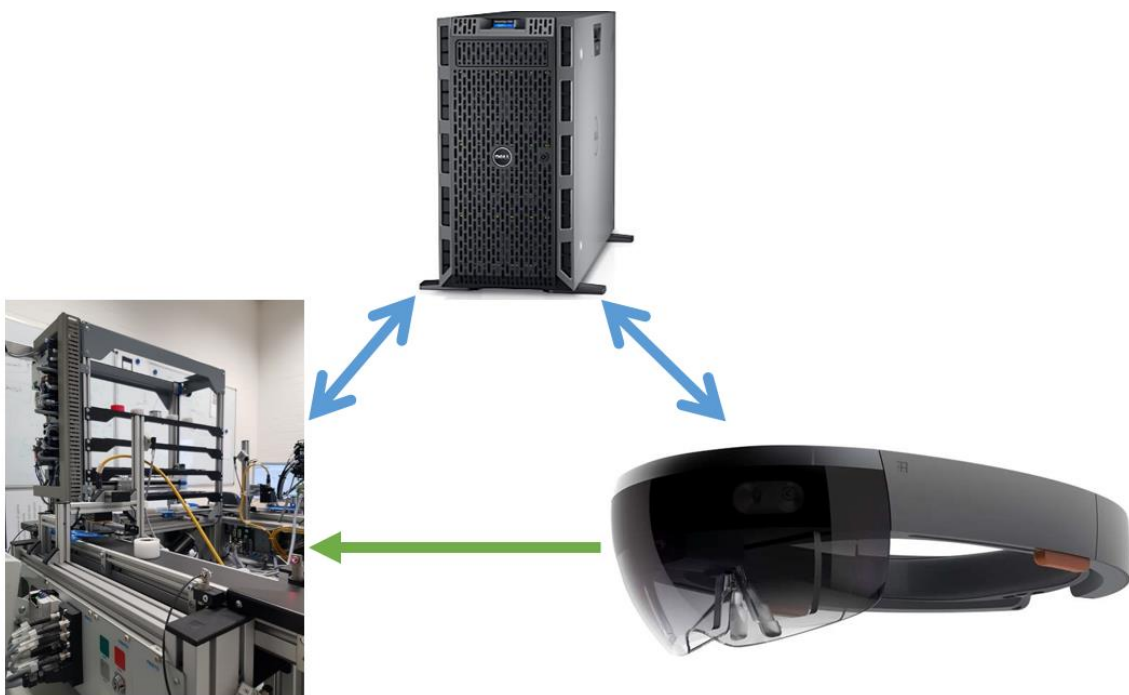


Abbildung 26: Überblick über das Gesamtszenario. Blau: Kommunikation Grün: Erfassung

6.2 Technologien

Mit der Entscheidung die HoloLens als Zielplattform für die Implementierung des Prototypens auszuwählen, wurde sich gleichzeitig an einige Technologien gebunden, welche in diesem Abschnitt kurz vorgestellt werden sollen.

Die Grundlage einer jeden HoloLens Anwendung ist die Spiel-Engine²⁴ Unity, welche zu Beginn dieses Abschnittes vorgestellt wird. Einhergehend mit Unity geht die Bindung an die (Skript-) Programmiersprache C#. Des Weiteren musste für die Objekterkennung ein Framework gewählt werden, welches mit Windows UWP²⁵ kompatibel ist (in diesem Fall Vuforia). Für die Kommunikation mit der Modelfabrik wird auf eine am Lehrstuhl vorhandene Modelfabrik API zurückgegriffen.

6.2.1 Unity

Unity ist der Name für eine Entwicklungsumgebung (Unity Editor) und gleichzeitig für die gleichnamige Computerspiel-Engine der Firma Unity Technologies. Sie gehört neben der Unreal und CryEngine zu den am häufigsten verwendeten 3D-Engines, welche auch kostenlos (abhängig vom Einsatzzweck) für Entwickler verfügbar sind. Unity erreicht ca. 770 Millionen Spieler Weltweit [Unit00].

Obwohl es sich bei Unity in erster Linie um eine Spiele-Engine handelt, wird diese auch außerhalb der Computerspielbranche eingesetzt. Neben der Möglichkeit diese für aufwändige 3D Simulationen zu nutzen, wird sie insbesondere noch für Anwendungen im Bereich VR, AR, MR eingesetzt.

Unity selbst bringt bereits eine Vielzahl von Features für die Entwicklung von MR-Anwendungen, Computerspielen und anderen 3D Anwendungen mit sich, unter anderem eine eigene Physik Engine. Vom Prinzip her handelt es sich bei Unity aber um kein professionelles Programm zur Modellierung von 3D Objekten. Dies ist zwar prinzipiell möglich mit Unity, da die Möglichkeiten zum Erstellen von einfachen Grundobjekten (Quader, Kugel etc.) mitgeliefert werden, aber für komplexere Modelle sollte auf Tools zurückgegriffen werden, welche insbesondere hierfür ausgelegt sind (z.B. Blender oder Rhinoceros) [Seif14]. Da im Rahmen dieser Arbeit die Modellierung von komplexeren

²⁴ Ein spezielles Framework für Computerspiele, welches insbesondere für die visuelle Darstellung verantwortlich ist.

²⁵ Universal Windows Plattform (kurz UWP) bezeichnet eine Laufzeitumgebung um Applikationen unter Windows 10 auszuführen.

3D Modellen nicht im Vordergrund steht, sondern lediglich ein Prototyp entwickelt wird, wurde auf die Verwendung einer weiteren Software für die 3D-Modellentwicklung verzichtet und auf die Primitives (so werden die 3D Grundobjekte in Unity bezeichnet) zurückgegriffen.

Die folgenden Abschnitte werden kurz in die Entwicklungsumgebung Unity einführen. Bevor diese jedoch vorgestellt wird, werden zunächst einige Begrifflichkeiten eingeführt:

- Scene: Jede Szene bezeichnet ein eigenes Level bzw. einen Abschnitt in einem Spiel oder einer Anwendung.
- GameObject: Hierbei handelt es sich um die Kernelemente in Unity, da sich nahezu alle Typen und Klassen hieraus ableiten [Tuli14]. Jedes 3D-Objekt und auch alle anderen Objekte innerhalb einer Szene (z.B. die Kamera²⁶). Jedes GameObject ist vergleichbar mit der Oberklasse Object in Java oder dem „System.Object“ im .NET Framework.
- Komponente: Jedes GameObject in Unity besteht mindestens aus der „Transform“ Komponente. Diese beschreibt die Position des GameObjects im Raum (Position, Rotation und Scale). Darüber hinaus können Objekte noch weitere Komponenten enthalten, z.B. das Verhalten oder eines GameObjects. Darüber hinaus kann das Game Object um Skript Komponenten erweitert werden.
- Frame: Ein Frame ist ein Update des Anzeigebildschirms. Die durchschnittliche Frame Rate der HoloLens liegt zwischen 30 und 60 FPS.

Die Entwicklungsumgebung von Unity ist der eines 3D-Animationsprogrammes nachempfunden und untergliedert sich in die folgenden Kernbereiche (siehe auch Abbildung 27) [Seif14, K.2.2]:

1. Scene View: Hier kann die aktuelle Szene und MR-Welt gestaltet werden. Optional kann auch ein Game View eingeblendet werden, welcher eine Vorschau auf die fertige Anwendung bietet (dies ist für MR Anwendungen nur begrenzt sinnvoll, da die physikalische Realität nicht mit abgebildet werden kann)
2. Hierarchy: Hier wird die Struktur aller existierenden GameObjects angezeigt. Per Drag&Drop können hier auch einfache Vererbungsbeziehungen erzeugt werden, in Unity auch „Parenting“ genannt. Bedeutet, dass Objekte gruppiert werden

²⁶ Die Kamera ist für die Blickwinkel auf die Szene aus Sicht des Nutzers zuständig. Für MR-Anwendungen wird auf die MR-Kamera zurückgegriffen, welche Teil des MRT ist.

können. Das oberste Objekt wird hierbei als „parent object“ und alle untergeordneten als „child objects“ bezeichnet. Änderungen des parent objects (z.B. der Position im Raum) wirken sich dann zwangsweise auch auf die child objects aus.

3. Inspector: Hier werden die Komponenten des aktuell ausgewählten GameObjects angezeigt (in diesem Fall des Quaders in Abbildung 27). Per Drag&Drop können diese weiteren Komponenten (zum Beispiel ein C#-Skript zugewiesen werden).
4. Project Browser: Dient zum Anzeigen und Verwalten aller digitalen Inhalte (in Unity auch Assets genannt). Hier können komplexere 3D-Objekte aus anderen Programmen oder Framework Parts gespeichert werden.

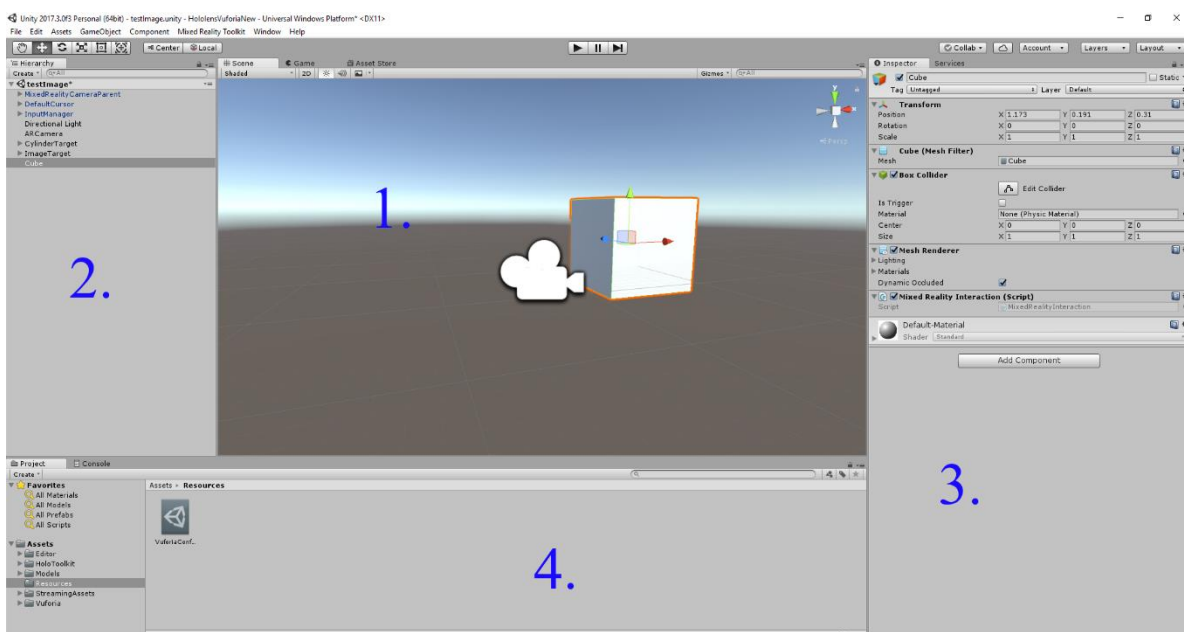


Abbildung 27: Ausschnitt aus der Unity 2017.3 Entwicklungsumgebung [eigene Aufnahme]

Unity selbst arbeitet sehr eng mit Microsoft zusammen und ist unter anderem Mitglied in der von Microsoft gegründeten .NET Foundation²⁷ und ermöglicht es Skripte in C# zu schreiben. Daneben ist es noch Möglich Unity Script (ähnlich wie Java Script) oder Boo zu verwenden.

²⁷ Unabhängige Organisation mit dem Ziel die Open Source Entwicklung rund um das .NET Framework zu verbessern und zu fördern.

6.2.2 C#

C# (ausgesprochen C sharp) wurde als (Skript-) Sprache im Rahmen dieser Arbeit ausgewählt, da diese aufgrund ihrer großen Ähnlichkeiten zu Java leicht zu erlernen war und da das Mixed Reality Toolkit ebenfalls alle Skripte in C# enthält.

C# wurde von Anders Hejlsberg²⁸ für Microsoft entworfen und erstmalig 2001 veröffentlicht. Ziel von Microsoft war es eine eigene objektorientierte Programmiersprache für das hauseigene .NET Framework zu entwickeln [Gunn00].

In Unity wird C# insbesondere als Skriptsprache eingesetzt um das Verhalten von GameObjects zu beschreiben. Eine Besonderheit bei den C# Skripten ist, dass jedes Skript von MonoBehaviour, welches eine Ableitung des GameObjects ist. MonoBehaviour beschreibt hierdabei das Kernverhalten eines jeden GameObjects, insbesondere durch die folgenden beiden Methoden (siehe auch: [Tech00]):

- Start(): Eine Methode die einmalig in der Lebenszeit des Skripts ausgeführt wird, wenn dieses aktiviert ist, bevor die Update Methode ausgeführt wird. Dies wird später zum Beispiel verwendet um einem erzeugten GameObject zuzuweisen auf welche Custom Messages es empfangen soll (siehe Abschnitt 6.2.3.1).
- Update(): Diese Methode bei der Aktualisierung von jedem Frame aufgerufen. Für MR-Anwendungen könnte hier die Bewegung eines Objektes implementiert werden.

6.2.3 Mixed-Reality Toolkit

Das Mixed Reality Toolkit (kurz MRT) ist eine Sammlung von Skripten, Assets, vorgefertigten Szenen etc., welche in Unity importiert werden können. Das MRT wurde von Microsoft zunächst unter dem Namen HoloToolkit speziell für die HoloLens veröffentlicht und später in MRT umbenannt. Mit dieser Umbenennung wurde es auch um den Support für weitere Windows MR-Headsets erweitert (hierbei handelt es sich insbesondere um VR Headsets [Stot17]).

Das MRT ist auf GitHub²⁹ verfügbar und unterteilt sich in die folgenden APIs:

²⁸ Dänischer Programmierer und Software-Architekt welcher u.a. Turbo Pascal entwickelte und später bei Microsoft im Bereich der DOS/Windows-Entwicklung arbeitete. Er ist u.a. Mitbegründer des .NET Frameworks.

²⁹ Siehe: <https://github.com/Microsoft/MixedRealityToolkit-Unity>.

1. Input: Umfasst Methoden rund um die Gesteneingabe. Hiermit können Eingaben innerhalb der MR erfasst werden. Der Augmented Operator könnte dies nutzen um mithilfe von virtuellen Buttons einen Fabrikpart zu steuern.
2. Spatial Mapping: Dies ist der Name für die Umwelterkennung und Lösung des SLAM Problems mit der HoloLens. Die Skripte in dieser API umfassen Methoden um die digitale und die physikalische Welt miteinander in Beziehung zu bringen. Ein Beispiel ist „TapToPlace“. Dieses Skript erlaubt es dem Anwender ein bereits erzeugtes MR-Objekte in der MR zu verschieben und an einem physikalischen Objekt zu platzieren (Beispielweise ein virtuelles Bild an einer realen Wand). Im Unterschied zum behandelten Problem dieser Arbeit können aber nur Objekte, welche bereits erzeugt wurden verschoben werden, das heißt es eignet sich nicht zur Objekterkennung, kann aber als Hilfsmethode zum Platzieren von statischen Objekten verwendet werden.
3. Spatial Understanding: Vorgefertigte Methoden zur Umweltauswertung oder auch zur Veränderung des Spatial Mappings. Es sind auch Methoden zur Umweltauswertung auf Basis des erzeugten Meshes³⁰ des Spatial Mappings vorhanden. So kann anhand der Form z.B. eine Couch im Raum identifiziert werden.
4. UX Controls: Hilfsmethoden zur Verbesserung der User Experience
5. Spatial Sound: Skripte um Spatial Sound (engl. für räumlicher Sound) zu integrieren.
6. Build, Boundary, Utilities: Umfassende Hilfsmethoden rund um die Erstellung und das Debuggen von MR-Anwendungen.
7. Sharing: Methoden, welche die Kommunikation mit anderen MR Devices, aber auch Servern und Datenbanken erlauben. Da diese Bibliotheken einen großen Teil der Implementierung ausmachen wird in Abschnitt 6.2.3.1 hierauf noch genauer eingegangen.

Für die Implementierung des Prototyps sind insbesondere die Bibliotheken rund um das Spatial Mapping, sowie Input und Sharing von Bedeutung.

³⁰ Mesh bezeichnet in der Computergrafik ein Polygonnetz zur Modellierung von 3D-Objekten.

6.2.3.1 Sharing

Die Sharing Library ermöglicht Anwendungen welche sich über mehrere Geräte erstrecken und erlaubt ebenfalls die Kooperation innerhalb der MR. Ursprünglich wurde diese für die Zusammenarbeit mit der NASA zur Planung des Mars Rovers entwickelt.

Eine besondere Stärke dieser Bibliothek ist die Kooperation innerhalb der MR, das heißt, dass eine MR-App auf mehreren HoloLens laufen kann und eine Änderung der MR auf der einen HoloLens direkt auf die weiteren Devices übertragen wird. Dies wird zwar im Rahmen dieser Arbeit nicht weiterverfolgt, bietet aber insbesondere für industrielle Anwendungen einen großen Nutzen, wenn eine Wartung zum Beispiel von 2 Akteuren vorgenommen wird.

Die kompletten Sharing Klassen wurden in portablen C++ Code geschrieben, aber mithilfe von SWIG³¹ auch in weitere Programmiersprachen übersetzt. Unterstützte Programmiersprachen, Plattformen und Chip Architekturen der Sharing Bibliotheken sind:

- C++, C#, Java
- Win32, UWP, OSX
- X86, x64, ARM

Für die Nutzung des Sharing Services muss auf einem Gerät ein Sharing Server gestartet werden und dort eine Sharing Session gestartet werden. Alle weiteren Geräte können sich dann auf diesen Server verbinden und an der Session teilnehmen.

6.2.4 Vuforia

Vuforia ist ein bekanntes AR Framework, welches direkt in Unity eingebunden werden kann. Ursprünglich war Vuforia insbesondere auf mobile AR-Anwendungen ausgelegt, später wurde jedoch eine umfassende Partnerschaft mit Microsoft geschlossen. Vuforia ergänzt die HoloLens insbesondere um visuelle Tracking Verfahren wie Marker- und Objekterkennung.

Auf Methoden des Vuforia Frameworks soll im Rahmen der Implementierung für die visuelle Erkennung der Werkstücke bzw. eines am Werkstück angebrachten Merkmales, zurückgegriffen werden, da die Tracking Methoden der HoloLens eher für die

³¹ Simplified Wrapper and Interface Generator ist ein Programmierwerkzeug welches es C/C++ geschriebene Module in andere Programmiersprachen übersetzt.

Umwelterkennung von größeren statischen Objekten ausgelegt ist. Weitere Informationen hierzu und insbesondere eine Auswertung über die Objekterkennung sind im Kapitel 7.1.2 zu finden.

6.2.5 Modelfabrik API

Bei der Modelfabrik API handelt es sich um eine in Java geschriebene Schnittstelle, welche eine Kommunikation zu Speicherprogrammierbaren Steuerungen (kurz SPS) von Siemens erlaubt (insbesondere der S7-300). Neben den Grundfunktionalitäten, welche eine Verbindung zur SPS herstellen, deren aktuellen Betriebsstatus abfragen können oder beliebige Speicherstellen auslesen oder beschreiben können, bietet sie insbesondere Funktionen an, welche speziell für die am Lehrstuhl vorhandene Modelfabrik entwickelt wurden. Somit ist es möglich ohne jegliche Kenntnisse der SPS Programmierung beispielsweise die verschiedenen Förderbänder zu kontrollieren (Starten/Stoppen/Richtung/Geschwindigkeit), sowie den Ein- und Auslagerprozess zu starten/stoppen und insbesondere sämtliche aktuellen Werte der Sensorik auszulesen. Hierzu zählen unter anderem alle Lichtschranken, sowie die 4 RFID Köpfe [Lawr16].

Für den Prototyp wird insbesondere die Leserichtung verwendet um die Werte aus der Sensorik auszulesen, insbesondere die Informationen des RFID Reader und der Lichtschranken. Erstere werden dafür benötigt ein Werkstück eindeutig zu identifizieren und letztere zur Definition der Strecke.

6.3 Implementierung des Prototyps

Nachdem in den vorherigen Abschnitten die grundlegenden Technologien für den konkreten Prototyp eingeführt wurden, beschäftigt sich dieser Abschnitt mit der konkreten Implementierung. Basierend auf dem in Kapitel 5 eingeführten Lösungskonzept des PositionMapper Patterns verfolgt die Implementierung grob gesagt die folgende Lösungsstrategie:

1. Objekt lokalisieren: die HoloLens (als hier gewählte MR-Hardware) detektiert ein neues physikalisches Objekt im Bildraum.
2. Objekt Typ bestimmen: Anschließend nimmt die HoloLens mithilfe eines visuellen Trackingverfahrens die Positionsbestimmung vor, sowie die Bestimmung des Typens des physikalischen Objektes (beispielweise zylinderförmiges Werkstück).

3. Abhängig von der bestimmten Position und des Objekttypens holt sich der Position Mapper weitere Informationen ein. Konkret handelt es sich um die Informationen darüber, ob bzw. wie viele Werkstücke sich innerhalb einer Strecke befinden. Eine Strecke wird dabei durch die Synchronisationspunkte (die RFID Reader) definiert.
4. Mithilfe der Informationen der Strecke und des Verhaltens wird das Werkstück eindeutig identifiziert.
5. Das MR-Objekt wird erzeugt und dem Endnutzer auf der HoloLens nahe dem getrackten Werkstück visualisiert.

Das in Abbildung 28 dargestellte Verteilungsdiagramm zeigt noch einmal die Verteilung der unterschiedlichen Programmteile des Prototyps und auf welchem Device sie letztendlich laufen.

Da das Rahmenprogramm der Modelfabrik im Rahmen dieser Arbeit nicht verändert wurde und lediglich auf die Modelfabrik API zurückgegriffen wurde, wird auf dieses nicht weiter eingegangen, für genauere Informationen hierzu sei auf [Lawr16] verwiesen.

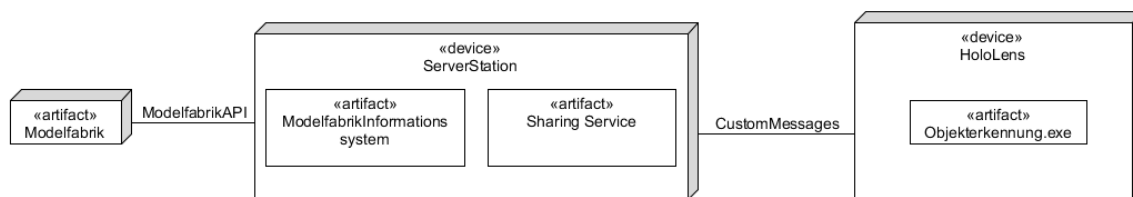


Abbildung 28: Verteilungsdiagramm für den Prototyp

6.3.1 Modelfabrikinformationssystem

Das Modelfabrikinformationssystem, welches für die Prototypen verwendet wird, stellt im Grunde nur eine Erweiterung des vorangegangenen Informationssystems aus [Lawr16, S.57] dar. Damals wurde ein Informationssystem implementiert, welches mit Hilfe der vier RFID Reader die Werkstücke im Umlauf gezählt hatte. Hierfür wurden die RFID Chips der einzelnen Werkstücke mit den Nummern 1-20 beschrieben und sobald ein Werkstück von einem RFID Reader erstmalig getrackt wurde, wurde ein Counter gestartet, welcher an den darauffolgenden Werkstücken wieder zurückgesetzt wurde. Falls ein Werkstück entfernt wurde, lief der Counter für dieses Werkstück ab und die Anzahl der Werkstücke im Umlauf wurde um dieses verringert.

Die damaligen verwendeten Counter (in der Arbeit und im Quellcode „WorkpieceCounter“) bildeten bereits in einen Vorläufer des Verhaltens aus dem Position Mapper Patterns (vgl. Position Mapper Pattern) ab. Für den Prototyp dieser Arbeit wird

ebenfalls auf die WorkpieceCounter zurückgegriffen um das Verhalten abzubilden. Allerdings wurden die Intervalle der Counter an die Strecken angepasst. Die durchschnittliche Geschwindigkeit eines Werkstückes lässt sich zu ungefähr 0,094m/s bestimmen und die Streckenlängen der Strecke 1 und Strecke 3 zu 0,66m. Strecke 2 ist ungefähr 0,35m lang. Darüber hinaus wird nun die Information verwendet, welcher RFID Reader ein Werkstück erkannt hat, da diese einen Synchronisationspunkt (vgl. Position Mapper Pattern) darstellen und die Abschnitte einer Strecke darstellen (siehe Abbildung 29). Die Strecke im Rahmen des Prototypens ist in nur drei Abschnitte eingeteilt wurden, da im nicht definierten Abschnitt neue physikalische Objekte aus dem Hochregallager ausgelagert werden können, diese aber erst am ersten Synchronisationspunkt (RFID 1) identifiziert werden können.

6.3.2 Sharing Service

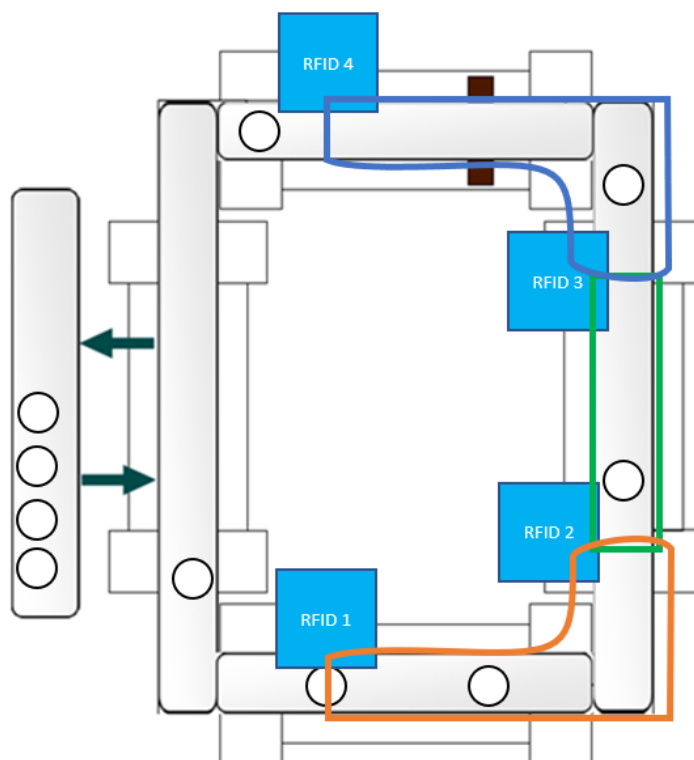


Abbildung 29: Aufbau der Modellfabrik inklusive definierter Strecken.
Orange: Strecke 1. Grün: Strecke 2. Blau: Strecke

Die Informationen des Modellfabrikinformationssystems werden mithilfe der Sharing Bibliothek des MRT an die HoloLens übertragen. Hierzu wurden die mit SWIG erzeugten Java Bibliotheken in das Informationssystem mit eingebunden und per Custom Message an die HoloLens gesendet. Hierfür muss allerdings zunächst der Sharing Service auf einem Rechner (in diesem Fall derselbe, wie der des Modellfabrikinformationssystems)

installiert und konfiguriert werden (eine Anleitung hierfür befindet sich auf der DVD im Anhang). Anschließend können sich alle Clients (Modelfabrikinformationssystem und die HoloLens) auf diesen verbinden und einer gemeinsamen Session beitreten. Innerhalb einer Session bestehen dann verschiedene Möglichkeiten des Informationsaustausches, wobei im Rahmen dieser Arbeit nur die Custom Messages verwendet werden. Diese ist hierbei immer wie folgt aufgebaut:

1. Sender (1 Byte): Jeder Client, welcher sich auf einen Server verbindet erhält seine eigene eindeutige ClientID. Diese kann später genutzt werden um den Sender einer Nachricht zu identifizieren.
2. MessageType (1 Byte): Definiert den Nachrichten Typ (z.B. Byte/Integer/Float etc.), damit der Empfänger weiß, wie er diese decodieren muss.
3. Payload (n Bytes): Enthält den Nachrichteninhalt.

6.3.3 Objekterkennung

Dieser Abschnitt behandelt das komplette Anwenderprogramm welches sich auf der HoloLens befindet mit Unity erzeugt wurde. Es untergliedert sich in die Objekterkennung und den Position Mapper.

6.3.3.1 Objekttracking

Für das Objekttracking wurden verschiedene Möglichkeiten des Vuforia Frameworks evaluiert. Die Einbindung des genauen Tracking Verfahrens ist hierbei modular und kann ohne weiteren Aufwand problemlos ausgetauscht werden. Für weitere Informationen zu den einzelnen potenziellen Tracking Verfahren sei auf Kapitel 7.1.2 verwiesen, in welchen eine Evaluation zu diesen Verfahren vorgenommen wird.

Für den Prototyp wurde letztendlich auf ein Verfahren zurückgegriffen, welches visuelle Merkmale innerhalb eines Bildes bzw. Musters erkennt (siehe auch Kapitel 4.2.4.2 Merkmalbasiertes Tracking). Der Grund hierfür ist schlicht, dass die derzeitige Stabilität um einiges höher ist (insbesondere in der Umgebung der Modelfabrik) als bei einer reinen geometrischen Erkennung der Werkstücke (siehe auch Kapitel 7.1.2).

6.3.3.2 Position Mapper

Die Implementierung des Position Mappers empfängt die Custom Messages auf Seiten der HoloLens und speichert sie zwischen. Hier wird definiertes Verhalten der physikalischen Werkstücke hinterlegt und ausgewertet. Mithilfe dieser Informationen, die im Rahmen der Implementierung sehr einfach gehalten sind und nur aus einer durchschnittlichen Fortbewegungsgeschwindigkeit bestehen und der getrackten Position

des visuellen Objekttrackingverfahren, ermittelt der Position Mapper die Identität des Werkstücks. Das Vorgehen dabei ist genau wie im Ablaufdiagramm der Objekterkennung (siehe Abbildung 25) beschrieben. Zunächst holt er die Information ein wie viele Werkstücke sich innerhalb der Strecke befinden und anschließend berechnet er ausgehend von dem Verhalten und der Position die Identität des Werkstücks. Diese wird dann an dem Werkstück angezeigt, wobei die korrekte Positionierung bereits Part des Objekttracking Verfahrens in Vuforia ist und daher keine weitere Herausforderung darstellte.

7 Evaluation

Ausgehend von der Implementierung des Prototyps und dem Entstehungsprozess dieses, werden in diesem Abschnitt zum einen die Möglichkeiten der unterschiedlichen Objekterkennungsverfahren evaluiert und abschließend eine Auswertung des Position Patterns aus Kapitel 5 Lösungskonzept vorgenommen.

7.1 Objekterkennung

Im Entstehungsprozess des Prototypens wurden von Beginn an die unterschiedlichsten Möglichkeiten des Objekt Trackings getestet. Da bereits bei Beginn der Arbeit fest stand, dass der Prototyp letztendlich auf der HoloLens implementiert werden sollte, wurden insbesondere Möglichkeiten rund um diese Hardware evaluiert.

7.1.1 Mithilfe der HoloLens

Zu Beginn des Entstehungsprozesses wurden insbesondere Möglichkeiten der Objekterkennung erörtert, welche auf die verfügbare Hardware der HoloLens zurückgreifen und nicht auf eine visuelle Auswertung. Hierbei wurde insbesondere die Möglichkeit der Tiefenkamera (auch bekannt als TOF³²-Kamera) diskutiert. Diese misst ähnlich wie die in Kapitel 4.2.3.2 vorgestellten Sensoren die Laufzeit eines Signales, in diesem Fall des Lichtes, die es bis zum Auftreffen auf ein Signal und wieder zurück benötigt. Diese ist direkt proportional zur Distanz.

In der HoloLens ist die Tiefenkamera aber insbesondere dafür eingebettet wurden, um das SLAM Problem zu lösen und um ein Abbild ihrer statischen Umwelt zu machen. Dies macht sich zum einen dadurch bemerkbar, dass die Kamera nur alle 0,2 Sekunden ihre Umwelt scannt und daher keine zu schnellen Bewegungen erfassen kann und zum anderen, dass die Auswertung der Sensordaten direkt mit dem Spatial Mapping der HoloLens zusammenhängen und von der Auflösung zu grob für die Erkennung der Werkstücke der Modelfabrik ist. Die Abbildung 30 zeigt beispielhaft, wie die HoloLens ihre Umwelt wahrnimmt. Auf der linken Seite ist dort die Modelfabrik zu sehen und deutlich zu erkennen, dass die Auflösung des Sensors derzeit noch zu grob ist um einzelne Objekte auf den Förderbändern zu erkennen. Dies liegt insbesondere auch daran, dass sich viele Störgrößen (wie Kabel, Sensoren etc.) nahe den Förderbändern befinden und

³² Time of Flight Kamera bezeichnet ein Kamerasysteme, welche mithilfe von Laufzeitmessverfahren die Distanz messen.

dadurch eine Vielzahl von Details wahrgenommen werden. Für größere Objekte hingegen ist eine Erkennung möglich. Ein Schreibtischstuhl, Tisch oder Computerbildschirm in Abbildung 30 könnte mithilfe des MRT (insbesondere Spatial Understanding) identifiziert werden.



Abbildung 30: Ausschnitt des Spatial Mappings der HoloLens aus dem Labor 106 des SSEs. Links ist die Modelfabrik zu sehen und rechts die Arbeitsplätze.

7.1.2 Visuell mithilfe von Vuforia

Da die Werkstücke der Modelfabrik mithilfe der Tiefenkamera nicht erfolgreich getrackt werden konnten, wurde auf ein visuelles Tracking Verfahren zurückgegriffen. Wie bereits erwähnt wurde hierfür das AR-Framework Vuforia ausgewählt da zum einen die Kompatibilität mit der HoloLens (insbesondere Windows UWP) vorhanden ist und zum anderen da Vuforia über eine Vielzahl unterschiedlicher visueller Tracking Verfahren verfügt. Im Rahmen dieser Arbeit wurden ausgewählte Verfahren getestet, welche insbesondere das Ziel verfolgt haben, dass das Werkstück möglichst wenig bis gar nicht modifiziert werden muss. Darüber hinaus sollten keine Verfahren verwendet werden, welche die Identität des Werkstückes direkt beinhalten (z.B. durch einen Marker, QR- oder Barcode), da dies in industriellen Anwendungen wenig sinnvoll wäre und nicht jedes Objekt mit Markern, Codes oder sonstigen beklebt werden soll.

7.1.2.1 Object Target

Zu Anfangs wurde ein allgemeines Objekterkennungsverfahren genannt Object Target in Vuforia evaluiert. Hierbei besteht die Möglichkeit ein Model entweder manuell

einzuscannen oder als CAD Datei zu hinterlegen. Eine Erkennung anhand eines in Catia³³ angelegten CAD Models eines Werkstückes war allerdings nicht erfolgreich, was vermutlich auf die Objektgröße zurückzuführen ist.

Die andere Möglichkeit besteht darin, dass Objekt mithilfe einer von Vuforia bereitgestellten Android App einzuscannen. Die App identifiziert hierbei bestimmte Merkmalspunkte anhand derer das Objekt später wieder identifiziert werden kann. Je mehr Punkte gefunden werden, desto stabiler funktioniert am Ende wieder die Objekterkennung. Im Rahmen dieser Arbeit wurde ein Werkstück, ein modifiziertes Werkstück, sowie eine Spielfigur aufgenommen und evaluiert (siehe Abbildung 31). Abbildung 31 ist deutlich zu erkennen, dass die Anzahl der detektierten Punkte vom Detailgrad des Objektes abhängt. Das Werkstück in seiner ursprünglichen Form lässt sich an gerade einmal 40 Merkmalspunkten festmachen, während das modifizierte Werkstück bereits 133 Punkte enthält. Die Actionfigur (Bowser) hingegen enthält sogar 311 Punkte.

Alle drei Objekte konnten mithilfe dieser Methode visuell getrackt werden. Allerdings ist die Stabilität stark abhängig von der Punkteanzahl und die beiden Werkstücke konnten lediglich in Umgebungen getrackt werden, welche sich deutlich von dem Werkstück unterscheiden (am besten auf einer Tischplatte ohne weitere Gegenstände in der Nähe, welche die Sicht einschränken könnten). Darüber hinaus muss die Kamera sich bei diesen beiden Objekten recht nahe (<1m) am Werkstück befinden. Aufgrund dieser beiden Nachteile eignet sich das Verfahren derzeit noch nicht wirklich für den Prototyp, da die Stabilität nicht garantiert werden kann.

Die Actionfigur hingegen lässt sich gut wieder im Raum Bildraum tracken, ebenfalls über etwas größere Entfernungen und in detailreichen Umgebungen.

³³ CAD System der französischen Firma Dassault Systemes, welches insbesondere in der Automobilbranche einen Standard darstellt.

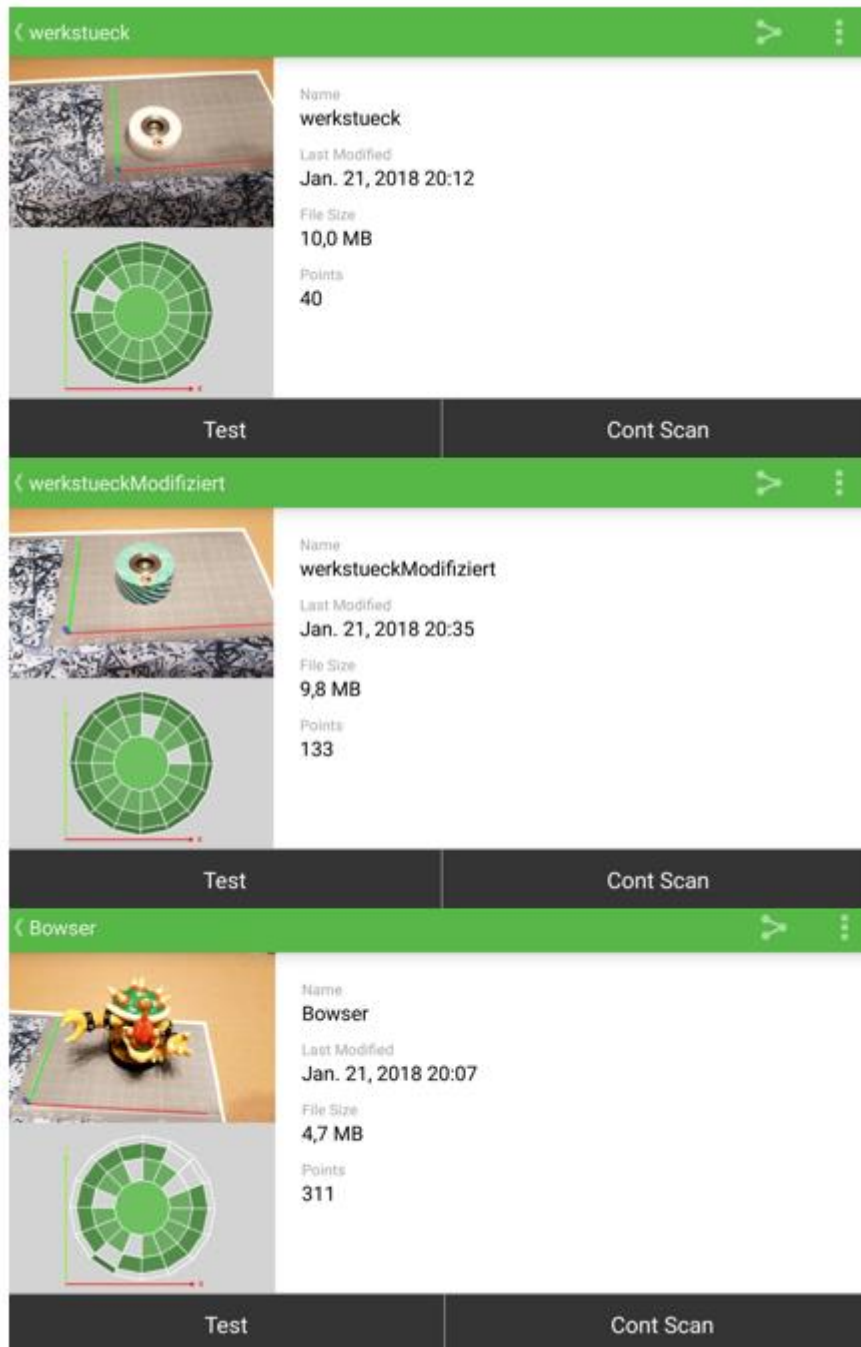


Abbildung 31: Ergebnisse des Model Target Scanners

7.1.2.2 Image Target

Da die Werkstücke mithilfe des Model Targets zwar identifiziert werden könnten, aber die Erkennung innerhalb der Modelfabrik doch noch sehr instabil ist, wurde für den Prototypen noch die Möglichkeit evaluiert, die Werkstücke anhand eines aufgeklebten Bildes zu identifizieren. Damit das Werkstück nicht anhand des Motives bereits von anderen Werkstücken unterschieden werden kann, wurde natürlich für alle Werkstücke dasselbe Bild verwendet und dient lediglich zur Stabilisierung des visuellen Trackings.

Die Stabilität des Trackingverfahrens hängt hierbei genau wie beim Objekt Target vom Detailreichtum des Bildes ab. Vuforia bietet Online bereits eine Möglichkeit das zu analysierende Bild zu bewerten (siehe Abbildung 32).




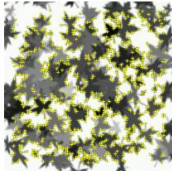
	Uploaded Image	Analized Image	Star Rating
Image with small number of features			★☆☆☆☆
Image with high number of features			★★★★★

Abbildung 32: Bild Bewertung des Vuforia Frameworks

Die Wertung besteht hierbei aus Sternen von 1-5 und hängt vom Detailgrad des zu analysierenden Bildes ab. Für die Prototypen wird ebenfalls das in Abbildung 32 zu erkennende untere Bild mit dem Laub verwendet.

Mithilfe des Image Targets wurde für den Prototyp eine zufriedenstellende Qualität erreicht. Instabilitäten welche hierbei noch auftreten sind insbesondere auf den Autofokus der Webcam der HoloLens zurückzuführen. Dadurch, dass der Endbenutzer selbst keine Möglichkeit hat, den Fokus einzustellen, wird ein Objekt in Bewegung manchmal erst erkannt, nachdem es von der Webcam fokussiert wurde.

7.2 Lösungskonzept

Das Tracking Verfahren selbst bietet noch keine Möglichkeit der Objekterkennung, da für diese das Objekt noch eindeutig identifiziert werden muss. Da die Objekte rein visuell nicht eindeutig identifiziert werden können, wurde im Kapitel 5 ein Lösungskonzept entwickelt, welches mithilfe weiterer Informationen und einer definierten Strecke das Objekt eindeutig identifizieren kann.

Durch die Implementierung des Prototypens auf der Modelfabrik wurde erfolgreich aufgezeigt, dass das Konzept funktioniert. Mithilfe der definierten Strecke und des Objektverhaltens konnten die Objekte eindeutig identifiziert werden. Dabei wurde mithilfe der HoloLens und Vuforia das Objekt anhand seines Typs visuell getrackt. Visuell bedeutet in diesem Fall an den Merkmalen des Image Targets. Mithilfe des Verhaltens und den vordefinierten Strecken konnte das Objekt dann anschließend immer

eindeutig identifiziert werden. Zu erwähnen ist hierbei, dass das Verhalten in seiner derzeitigen Form noch sehr einfach gehalten ist und nicht auf äußere Einflüsse reagiert. Das heißt, sollte beispielsweise ein Förderband angehalten werden, bekommt das Verhalten dies derzeit nicht mit und würde weiterhin davon ausgehen, dass das Werkstück sich weiterbewegt. Ein erfolgreicher Prozess dieses Tracking Verfahrens ist in Abbildung 33 zu erkennen. Hierbei handelt es sich um eine Aufnahme des Objekterkennungsprogrammes auf der HoloLens. Mithilfe einer Funktion der HoloLens genannt Mixed Reality Capture wurde ein Ausschnitt aufgenommen, welcher einen Einblick ermöglicht wie ein Benutzer, welcher die HoloLens trägt, die Mixed Reality zu sehen bekommt.

Das Verhalten lässt sich allerdings erweitern und könnte beispielsweise durch eine zusätzliche Auswertung der Förderbandstatus auch auf solche Änderungen reagieren. Prinzipiell ist die Funktionsweise des Patterns also erfolgreich evaluiert wurden, seine Effizienz hängt aber insbesondere von der Modellierung des Verhaltens und der Definition der Strecke ab.

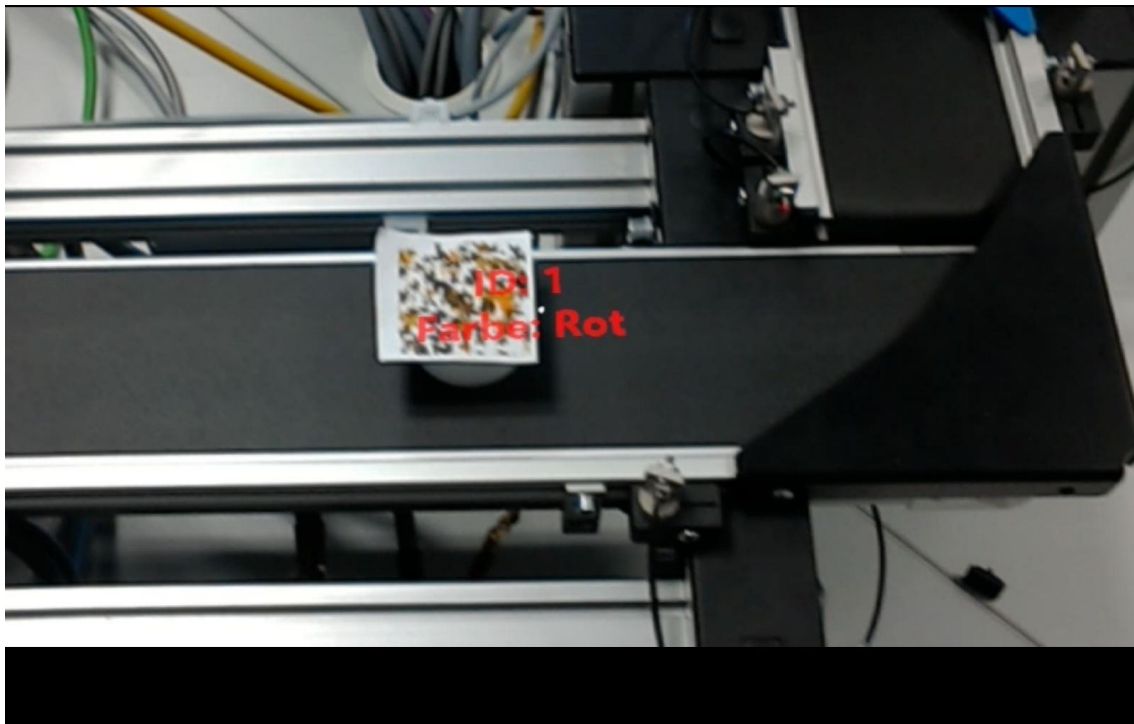


Abbildung 33: Mixed Reality Capture der Objekterkennung mithilfe der HoloLens

8 Fazit und Ausblick

Abschließend werden in diesem Kapitel noch einmal die Ergebnisse dieser Arbeit reflektiert und ein Ausblick auf potentielle weitere Arbeiten und Anwendungen gegeben.

8.1 Fazit

Ziel dieser Arbeit war es ein Konzept zu entwickeln, welches es ermöglicht physikalische Objekte zu erkennen und ihre dazugehörigen digitalen Informationen mithilfe einer MR-Anwendung an diesen zu visualisieren.

Hierfür wurden in Kapitel 2 zunächst die theoretischen Grundlagen erörtert, warum sich die Arbeitsumgebung des Menschen durch Industrie 4.0 verändern und inwiefern unter Zuhilfenahme aktueller Technologien (insbesondere MR-Technologien) veränderte Mensch Maschinen Schnittstellen ermöglicht werden können.

Anschließend wurden die Probleme herausgearbeitet auf welche ein Entwickler beim Entwurf einer MR-Anwendung stoßen könnte und insbesondere das Problem der Objekterkennung in den Fokus dieser Arbeit gestellt.

Darauffolgend wurde der aktuelle Stand der Forschung im Bereich AR und Objekterkennung genauer untersucht und auf Basis dieser Ergebnisse anschließend das Pattern des Position Mappers entworfen, welches es ermöglicht physikalische Objekte zu Beginn erst einmal vom Typ her zu erkennen und anhand ihrer getrackten Position und ihres Verhaltens eindeutig zu identifizieren.

Dieses Pattern wurde erfolgreich auf der HoloLens implementiert um die dynamischen Werkstücke der Modelfabrik zu identifizieren. Hierbei wurde insbesondere bei der Architektur darauf geachtet, dass das Tracking Verfahren und Verhalten als austauschbare Komponenten realisiert wurden.

Mit Hilfe des Prototypens wurde grundlegend die Funktionalität des Position Mapper Patterns bewiesen und aufgezeigt, dass insofern, dass Verhalten und die Strecke bekannt sind, sich dieses ebenfalls auf vergleichbare Probleme übertragen lässt.

8.2 Ausblick

Der Fokus dieser Arbeit wurde ab Kapitel 3 von einem allgemeinen MR-Framework für industrielle Anwendungen insbesondere auf die Objekterkennung verschoben. Vom Prinzip her bieten sich aber noch viele weitere Möglichkeiten im Bereich Mixed Reality

an. Unter anderem wäre es denkbar, den Prototypen weiter auszubauen und insbesondere auch um eine Eingaberichtung zu erweitern. Das heißt, dass mithilfe von virtuellen Buttons innerhalb der MR die Modelfabrik gesteuert werden könnte oder ähnliches.

Darüber hinaus muss einem bewusst sein, dass es sich bei der HoloLens derzeit noch um die erste Edition und insbesondere um eine Hardware für Entwickler handelt. Es ist damit zu rechnen, dass sich die Hardware im Bereich MR weiter verbessert und insbesondere auch, dass sich die visuellen Tracking Verfahren immer weiter verbessern. Daher wird es vermutlich zukünftig auch möglich sein, die Objekterkennung der Werkstücke ohne zusätzliche Hilfsmerkmale vorzunehmen. Überdies werden viele Objekte im industriellen Kontext um einiges größer sein und daher auch schon jetzt besser zu tracken.

Abschließend soll noch erwähnt werden, da bereits angekündigt wurde, dass die 2. Version der HoloLens über einen eigenen Prozessor für künstliche Intelligenz verfügen wird, dass sich Verfahren aus diesem Bereich Lösungsmöglichkeiten für den Position Mapper eignen könnten. Insbesondere das Verhalten von physikalischen Objekten könnte mit solchen Verfahren gelernt werden, anstatt dass es simulativ implementiert werden muss.

9 Anhang

9.1 Abbildungsverzeichnis

Abbildung 1: Überblick über die verschiedenen Stufen der industriellen Revolution [KaWH13 S. 17].....	6
Abbildung 2: Abbildung eines Siemens SIMATIC Panel PCs 877 [Indu00]	9
Abbildung 3: Prinzip der HMD-Projektion [Date17]	11
Abbildung 4: Mixed Reality Spektrum [00a].....	12
Abbildung 5: Abseitslinie im Fußball [Vide18].....	14
Abbildung 6: Verschmelzung einer realen Umgebung (links) und eines virtuellen Objektes (rechts) zur AR (Mitte) [DBGJ14 S. 242].....	15
Abbildung 7: Ausschnitt aus dem AR-Spiel Pokemon GO	16
Abbildung 8: Google Glass Explorer Edition (2014).....	17
Abbildung 9: Abbildung der Microsoft HoloLens [Micr00]	18
Abbildung 10: Unterschiede der drei Technologien aus der Sicht von Microsoft: In der AR ist das Bild im Vergleich nur MR nicht korrekt in die Umgebung eingepasst [Bare17]	19
Abbildung 11: B2B Anwendungsfelder für HMDs [EsOL16]	20
Abbildung 12: aktiver Spritzprozess im Projekt generative Fertigung	22
Abbildung 13: Bild der Modelfabrik der TU Clausthal	24
Abbildung 14: Übersicht über die unterschiedlichen Welten	26
Abbildung 15: Skizze des SLAM Problems (inspiriert von [Pfin13])	29
Abbildung 16: Übersicht über die Probleme innerhalb der verschiedenen Welten. Die blauen Probleme werden im Rahmen dieser Arbeit behandelt	30
Abbildung 17: Ortung durch Entfernungsmessung in zwei Dimensionen. Der Schnittpunkt der 3 Kreise der jeweiligen Abstände zum Satellit S1-S3 beschreiben die aktuelle Position des Empfängers. [Schü14].....	39
Abbildung 18: Beispiele für Marker [DBGJ14 Abb. 8.13].....	41
Abbildung 19: Zusammenhang zwischen Kamerakoordinatensystem C und dem Markerkoordinatensystem M.. [DBGJ14 S.260]	42
Abbildung 20: Übersicht über den Tracking Prozess mithilfe eines 3D Modelles	43
Abbildung 21: Abbildung eines Bar-Codes (links) sowie eines QR-Codes (rechts).	45

Abbildung 22: Position Mapper als Lösungsmethode für die Probleme der physikalischen Welt und der Mixed Reality	49
Abbildung 23: Struktur des Position Mapper Patterns	51
Abbildung 24: Zustandsdiagramm eines dynamischen physikalischen Objektes.....	52
Abbildung 25: Zustandsdiagramm eines Objekterkennungsvorgangs.....	53
Abbildung 26: Überblick über das Gesamtszenario. Blau: Kommunikation Grün: Erfassung.....	56
Abbildung 27: Ausschnitt aus der Unity 2017.3 Entwicklungsumgebung [eigene Aufnahme].....	59
Abbildung 28: Verteilungsdiagramm für den Prototyp.....	64
Abbildung 29: Aufbau der Modelfabrik inklusive definierter Strecken. Orange: Strecke 1. Grün: Strecke 2. Blau: Strecke	65
Abbildung 30: Ausschnitt des Spatial Mappings der HoloLens aus dem Labor 106 des SSEs. Links ist die Modelfabrik zu sehen und rechts die Arbeitsplätze.	69
Abbildung 31: Ergebnisse des Model Target Scanners	71
Abbildung 32: Bild Bewertung des Vuforia Frameworks	72
Abbildung 33: Mixed Reality Capture der Objekterkennung mithilfe der HoloLens.....	73

9.2 Tabellenverzeichnis

Tabelle 1: Merkmale von VR im Vergleich zu konventioneller Computergraphik [DBGJ14 S.14 Tab. 1.1	13
Tabelle 2: Unterschiede der drei Technologien aus der Sicht von Microsoft [Wasi17].20	

9.3 Literaturverzeichnis

- [Augm00a] *Augmented Reality*. URL <https://www.verlagederzukunft.de/augmented-reality/>. - abgerufen am 2018-03-18. — Verlage der Zukunft
- [Augm00b] *Augmented Reality - AR - Golem.de*. URL <https://www.golem.de/specials/augmented-reality/>. - abgerufen am 2018-03-19. — Golem.de
- [Augm00c] *Augmented Reality am OP-Tisch - Klinik - via medici*. URL <https://www.thieme.de/viamedici/klinik-faecher-chirurgie-1531.htm/a/augmented-reality-im-op-33591.htm>. - abgerufen am 2018-03-17. — Thieme Via medici
- [Av(a17)] *AV (augmented virtuality)*. URL <https://www.itwissen.info/AV-augmented-virtuality.html>. - abgerufen am 2018-03-19. — ITWissen.info
- [Azum97] AZUMA, RONALD T.: *A Survey of Augmented Reality*. URL <http://www.cs.unc.edu/~azuma/ARpresence.pdf>. - abgerufen am 2018-03-11
- [BaHV14] BAUERNHANSL, THOMAS ; HOMPEL, MICHAEL TEN ; VOGEL-HEUSER, BIRGIT: *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung · Technologien · Migration*. 2014. Aufl. Wiesbaden : Springer Vieweg, 2014 — ISBN 978-3-658-04681-1
- [Bare17] BAREL, AVI: *The differences between VR, AR &MR*. URL <https://uxplanet.org/the-differences-between-vr-ar-mr-27012ea1c5>. - abgerufen am 2018-03-25. — UX Planet
- [BeKo13] BETTENHAUSEN, KURT ; KOWALEWSKI, STEFAN: *Cyber-Physical Systems: Chancen und Nutzen aus Sicht der Automation*, 2013
- [Date17] *Datenhelm*. URL <https://www.itwissen.info/Datenhelm-head-mounted-display-HMD.html>. - abgerufen am 2018-03-19. — ITWissen.info
- [DBGJ14] DÖRNER, RALF ; BROLL, WOLFGANG ; GRIMM, PAUL ; JUNG, BERNHARD: *Virtual und Augmented Reality*. 2013. Aufl. Berlin Heidelberg : Springer Vieweg, 2014 — ISBN 978-3-642-28902-6
- [Dete00] *Detektion Definition > Begriff, Bedeutung, Erklärung > Wissenschaft*. URL <http://definition-online.de/detektion/>. - abgerufen am 2018-03-14
- [DGJS02] DUNEMANN, OLIVER ; GEIST, INGOLF ; JESSE, ROLAND ; SAAKE, GUNTER ; SATTLER, KAI-UWE: Informationsfusion auf heterogenen Datenbeständen. In: *Informatik Forschung und Entwicklung* Bd. 17 (2002), Nr. 3, S. 112–122
- [EsOL16] ESSER, RALF ; OPPERMAN, LEIF ; LUTTER, TIMM: *Head Mounted Displays in deutschen Unternehmen Ein Virtual, Augmented und Mixed Reality Check*, 2016

- [Eyet00] *EyeTap Personal Imaging Lab.* URL <http://www.eyetap.org/research/eyetap.html>. - abgerufen am 2018-03-20. — eyetap.org
- [Goog00] *Google Glass.* URL <http://www.rundum-technik.de/google-glass.html>. - abgerufen am 2018-03-20
- [Gunn00] GUNNERSON, ERIC: *C# - Rheinwerk openbook.* URL <http://openbook.rheinwerk-verlag.de/csharp/index.htm>. - abgerufen am 2018-04-02
- [Gyro00] *Gyrosensor :: gyro sensor :: ITWissen.info.* URL <https://www.itwissen.info/Gyrosensor-gyro-sensor.html>. - abgerufen am 2018-03-13
- [Howh00] *How HoloLens Sensors Work | The Imaginative Universal.* URL <http://www.imaginativeuniversal.com/blog/2015/10/31/how-hololens-sensors-work-2/>. - abgerufen am 2018-03-14
- [Indu00] *Industry Support Siemens.* URL [https://support.industry.siemens.com/cs/document/21497698/neuer-simatic-panel-pc-877-ab-sofort-lieferbar-\(lf\)?dti=0&lc=de-DE](https://support.industry.siemens.com/cs/document/21497698/neuer-simatic-panel-pc-877-ab-sofort-lieferbar-(lf)?dti=0&lc=de-DE). - abgerufen am 2018-03-18
- [Inst16] INSTANDHALTUNGSFORUM ; BIEDERMANN, H. (Hrsg.): *Lean smart maintenance: Konzepte, Instrumente und Anwendungen für eine effiziente und intelligente Instandhaltung: 30. Instandhaltungsforum, Praxiswissen für Ingenieure - Instandhaltung.* Köln : TÜV Media, 2016 — ISBN 978-3-7406-0096-9
- [Ivan00] *Ivan Sutherland - A.M. Turing Award Winner.* URL https://amturing.acm.org/award_winners/sutherland_3467412.cfm. - abgerufen am 2018-03-16
- [KaWH12] KAGERMANN, HENNING ; WAHLSTER, WOLFGANG ; HELBIG, JOHANNES: *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 (2012)*
- [Kolo16] KOLOKYTHAS, PANAGIOTIS: *Hololens: Microsoft lüftet HPU-Geheimnis.* URL <https://www.pcwelt.de/news/Hololens-Microsoft-lueftet-HPU-Geheimnis-10028103.html>. - abgerufen am 2018-03-20. — PC-WELT
- [LaSi00] LACKES, RICHARD ; SIEPERMANN, MARKUS: *Objektorientierung.* URL <http://wirtschaftslexikon.gabler.de/Definition/objektorientierung.html>. - abgerufen am 2018-02-28. — Gabler Wirtschaftslexikon
- [Lawr16] LAWRENZ, SEBASTIAN: *Entwurf einer objekt-orientierten Schnittstelle für eine SPS zur Realisierung eines cyberphysischen Systems (2016)*
- [LeSe15] LEE, EDWARD ASHFORD ; SESHIA, SANJIT ARUNKUMAR: *Introduction to embedded systems: a cyber-physical systems approach.* 2. ed., Printing 2.0. Aufl. Lulu : LeeSeshia.org, 2015 — ISBN 978-1-312-42740-2

- [MeRe11] MEHLER-BICHER, ANETT ; REIß, MICHAEL: *Augmented Reality: Theorie Und Praxis*. München : De Gruyter Oldenbourg, 2011 — ISBN 978-3-486-59837-7
- [Micr00] MICROSOFT: *Microsoft HoloLens / The leader in mixed reality technology*. URL <https://www.microsoft.com/de-de/hololens>. - abgerufen am 2018-03-20
- [MiKi94] MILGRAM, PAUL ; KISHINO, FUMIO: A Taxonomy of Mixed Reality Visual Displays. In: *IEICE Trans. Information Systems* Bd. E77-D, Nr. 12 (1994), S. 1321–1329
- [Mitd00] *Mit der Smart Factory auf dem Weg in die Produktion der Zukunft*. URL <https://www.maschinenmarkt.vogel.de/mit-der-smart-factory-auf-dem-weg-in-die-produktion-der-zukunft-a-413041/>. - abgerufen am 2018-03-18
- [Mixe00] *Mixed reality*. URL https://developer.microsoft.com/en-us/windows/mixed-reality/mixed_reality. - abgerufen am 2018-02-27
- [NGWS05] NIEHSEN, WOLFGANG ; GARNITZ, RAINER ; WEILKES, MICHAEL ; STÄMPFLE, MARTIN: Informationsfusion für Fahrerassistenzsysteme. In: MAURER, M. ; STILLER, C. (Hrsg.): *Fahrerassistenzsysteme mit maschineller Wahrnehmung*. Berlin/Heidelberg : Springer-Verlag, 2005 — ISBN 978-3-540-23296-4, S. 43–57
- [Oubb12] OUBBATI, MOHAMED: Einführung in die Robotik-Sensoren.
- [Pfin13] PFINGSTHORN, MAX: *Generalized Simultaneous Localization and Mapping (SLAM) on Graphs with Multimodal Probabilities and Hyperedges*, 2013
- [ReDr06] REITMAYR, GERHARD ; DRUMMOND, TOM: Going out: robust model-based tracking for outdoor augmented reality. In: : IEEE, 2006 — ISBN 978-1-4244-0650-0, S. 109–118
- [Reif16] REIF, K. (Hrsg.): *Grundlagen Fahrzeug- und Motorentechnik im Überblick: konventioneller Antrieb, Hybridantriebe, Bremsen, Elektrik und Elektronik, Bosch Fachinformation Automobil*. 2., überarbeitete Auflage. Aufl. Wiesbaden : Springer Vieweg, 2016 — ISBN 978-3-658-04961-4
- [Röb114] RÖBLER, PHILLIP: *Industrie 4.0: Der Arbeitsplatz von morgen – computer-automation.de*. URL <http://www.computer-automation.de/unternehmensebene/produktionssoftware/artikel/108139/2/>. - abgerufen am 2018-03-08
- [RuPu06] RUSER, HEINRICH ; PUENTE LEÓN, FERNANDO: Informationsfusion- Eine Übersicht (2006)
- [Schü14] SCHÜTTLER, TOBIAS: *Satellitennavigation, Technik im Fokus*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2014 — ISBN 978-3-642-53886-5
- [Seco17] *Second version of HoloLens HPU will incorporate AI coprocessor for implementing DNNs*. URL <https://www.microsoft.com/en-us/research/blog/second-version-hololens-hpu-will-incorporate-ai->

- coprocessor-implementing-dnns/. - abgerufen am 2018-03-20. — Microsoft Research
- [Seif14] SEIFERT, CARSTEN: *Spiele entwickeln mit Unity: 3D-Games mit Unity und C# für Desktop, Web & Mobile*. München : Carl Hanser Verlag GmbH & Co. KG, 2014 — ISBN 978-3-446-43939-9
- [Stot17] STOTT, LEE: *Mixed Reality Immersive— A Beginners guide to building for MR – Microsoft Faculty Connection*. URL https://blogs.msdn.microsoft.com/uk_faculty_connection/2017/10/09/mixed-reality-immersive-a-beginners-guide-to-building-for-mr/. - abgerufen am 2018-04-02
- [Surf00] SURF: *Speeded Up Robust Features*. URL <http://www.vision.ee.ethz.ch/~surf/>. - abgerufen am 2018-03-13
- [Suth65] SUTHERLAND, IVAN E.: *The Ultimate Display*. URL <http://www8.informatik.umu.se/~jwworth/The%20Ultimate%20Display.pdf>. - abgerufen am 2018-03-11
- [Tech00] TECHNOLOGIES, UNITY: *Unity - Scripting API*. URL <https://docs.unity3d.com/ScriptReference/index.html>. - abgerufen am 2018-04-02
- [Torg91] TORGE, WOLFGANG: *Geodesy*. 2. ed. Reprint 2015. Aufl. : De Gruyter, 1991 — ISBN 978-3-11-117454-9
- [Tuli14] TULIPER, ADAM: *Unity - Entwickeln Ihres ersten Spiels mit Unity und C#*. URL <https://msdn.microsoft.com/de-de/magazine/dn759441.aspx>. - abgerufen am 2018-04-02. — msdn.microsoft.com
- [Unit00] *Unity - Fast Facts*. URL <https://unity3d.com/de/public-relations>. - abgerufen am 2018-04-02
- [Wasi00a] *Was ist Industrie 4.0?* URL <http://www.plattform-i40.de/I40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html>. - abgerufen am 2018-03-18
- [Wasi00b] *Was ist eine Smart Factory?* URL <https://www.bigdata-insider.de/was-ist-eine-smart-factory-a-643838/>. - abgerufen am 2018-03-18
- [Wasi17] *Was ist Mixed Reality? – Unterschied zu Virtual & Augmented Reality erklärt*. URL <http://www.giga.de/extra/mixed-reality/>. - abgerufen am 2018-03-20. — GIGA
- [XuXu16] XU, GUOCHANG ; XU, YAN: *GPS: Theory, Algorithms and Applications*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2016 — ISBN 978-3-662-50365-2
- [Zhan03] ZHANG, J: *Angewandte Sensorik*.

Die beiliegende DVD beinhaltet:

- Diese Arbeit als PDF Dokument
- Das Unity Projekt für die HoloLens
- Das Java Projekt des Modelfabrik Informationssystems
- Die verwendete Version des Mixed Reality Toolkits
- Eine Anleitung wie der Sharing Service zu konfigurieren ist
- Ein Video welches die erfolgreiche Ausführung des Prototypens demonstriert